UNIVERSITY OF CALIFORNIA
RIVERSIDE

Advanced Optimization and Data-Driven Control in Smart Grid

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Wei Wang

March 2020

Dissertation Committee:

Dr. Nanpeng Yu, Chairperson
Dr. Daniel Wong
Dr. Shaolei Ren

The Dissertation of Wei Wang is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

First of all, I would like to express my gratitude to Prof. Nanpeng Yu, who gave me continuous support throughout my Ph.D. study and shared insightful knowledge in the area of power systems with me. Since I joined the Smart City Innovation Laboratory of University of California Riverside five years ago, Prof. Nanpeng Yu has guided me through various research challenges in the field of optimization and machine learning applications in the smart grid. I would have not been able to complete this dissertation without his collaboration and valuable insights.

Secondly, I would like to thank all of my lab colleagues, Jie Shi, Yuanqi Gao, Wenyu Wang, Farzana Kabir, and Brandon Foggo for the accompany throughout the five years. It is a very enjoyable experience to work with them. Their brilliant work has always inspired me with new ideas. New solutions are discovered by various discussion with them. I would also like to acknowledge all of my collaborators, Yang Liu, Amirali Abdolrashidi, Prof. Daniel Wong, and Yinglun Li. I am thankful for their knowledge in different domains and innovative works, which help me accomplish extra publications.

I am also grateful for all of my committee members, Prof. Matthew Barth, Prof. Zhiyun Qian, Prof. ShaoLei Ren, and Prof. Daniel Wong. I am thankful for their service during my preliminary exams and final dissertation defense. Their feedback and suggestions have helped me further improve my presentation skills and lead me to new thoughts on the research topics. I appreciate all the professors who have guided and taught me during my Ph.D. study. They have helped me establish a solid foundation and motivated me to explore new research problems. Their guidance has benefited not only my research but also

my future career.

Finally and most importantly, I would like to express my sincere gratitude to my family. I am grateful that they have provided me with the best living environment to grow up, the best education to sense the world, and the best care to keep me away from pressure and worries. They have always supported me in exploring any new possibility. I am so fortunate to have had my family as my strongest backup force so that I can pursue my dreams.

The content of this thesis is a re-print of the materials that are appeared in the following publications:

- W. Wang and N. Yu, "LMP decomposition with three-phase DCOPF for distribution system," in 2016 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia), Nov. 2016, pp. 1–8. (Chapter 2.2)

- W. Wang and N. Yu, "Chordal conversion based convex iteration algorithm for three-phase optimal power flow problems," IEEE Trans. Power Syst., vol. 33, no. 2, pp. 1603-1613, Mar. 2018. (Chapter 2.3)

- W. Wang and N. Yu, "Phase balancing in power distribution network with data center," SIGMETRICS Perform. Eval. Rev., vol. 45, no. 2, pp. 64-69, Oct. 2017. (Chapter 3.2)

- W. Wang, A. Abdolrashidi, N. Yu, and D. Wong, "Frequency regulation service provision in data center with computational flexibility," Applied Energy, vol. 251, p.113304, 2019. (Chapter 3.3)

- W.Wang, N. Yu, J. Shi, and Y. Gao, "Volt-VAR control in power distribution systems with deep reinforcement learning," in IEEE SmartGridComm, Oct. 2019, pp. 1-7. (Chapter 4.2)

- W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe off-policy deep reinforcement learning algorithm for Volt-VAR control in power distribution systems," IEEE Trans. Smart Grid, pp. 1-11, Dec. 2019. (Chapter 4.3)

- W. Wang and N. Yu, "A machine learning framework for algorithmic trading with virtual bids in electricity markets," in IEEE PES GM, Aug. 2019, pp. 1-8. (Chapter 5)

To my parents for all the support.

ABSTRACT OF THE DISSERTATION

Advanced Optimization and Data-Driven Control in Smart Grid

by

Wei Wang

Doctor of Philosophy, Graduate Program in Electrical Engineering
University of California, Riverside, March 2020
Dr. Nanpeng Yu, Chairperson

Power grids have been evolving rapidly over the last decade with widespread penetration of distributed energy resources such as distributed renewable generation resources. On one hand, the intermittency of these resources brought tremendous challenges to the operations and planning of the smart grid. On the other hand, the installation of advanced communication, monitoring, and remotely controlled devices enabled sophisticated control schemes. Thus, it is imperative to develop advanced optimization and data-driven controls to improve the economics and reliability of the system.

From the perspective of distribution system operators, it is critical to maximize social welfare while satisfying operational constraints. To better coordinate the operations of DERs and improve the efficiency of distribution systems, three-phase optimal power flow (OPF) algorithms such as DCOPF and ACOPF are developed. To remove the dependency on complete and accurate system models, deep reinforcement learning-based Volt-VAR control schemes are proposed to keep voltages of the distribution system within the allowable range.

From the perspective of end-use customers, it is critical to manage the flexible demand resources in order to minimize energy bills while satisfying their energy needs. By providing ancillary services, the flexible demands not only receive monetary compensations but also improve the reliability and stability of the power grids. We developed control schemes for data centers to provide the phase balancing service to power distribution grids and the frequency regulation service to transmission systems.

From the perspective of financial traders in electricity markets, it is critical to maximize their trading profits while managing the portfolio risk. A machine learning-based trading framework is proposed to identify a portfolio of profitable virtual bids in the U.S. wholesale electricity markets.

Contents

List of Figures **xiii**

List of Tables **xv**

1 Introduction **1**
1.1 Background . . . 2
1.1.1 Increasing Penetration of Distributed Energy Resources . . . 2
1.1.2 Emergent Computing Load . . . 3
1.1.3 Widespread Deployment of Smart Meters and Sensors . . . 4
1.1.4 Wholesale Electricity Market . . . 4
1.2 Technical Challenges and Research Opportunities . . . 5
1.2.1 Three-phase OPF in Distribution Systems . . . 6
1.2.2 Ancillary Services Provision with Data Centers . . . 7
1.2.3 Data-driven Volt-VAR Control . . . 8
1.2.4 Algorithmic Trading with Virtual Bids . . . 9
1.3 Contributions . . . 9
1.4 Organization . . . 13

2 Three-phase OPF in Distribution System **15**
2.1 Related Works . . . 15
2.2 Three-phase DCOPF . . . 17
2.2.1 Problem Formulation . . . 17
2.2.2 Simulation Results . . . 32
2.2.3 Conclusion . . . 35
2.3 Three-phase ACOPF . . . 36
2.3.1 Problem Formulation . . . 36
2.3.2 Technical Approach . . . 42
2.3.3 Numerical Study . . . 55
2.3.4 Conclusion . . . 63

3 Ancillary Service Provision with Data centers **65**
3.1 Related Works . . . 66
3.2 Phase Balancing in Power Distribution Network with Data Center . . . 68
3.2.1 Overall Framework . . . 68
3.2.2 Problem Formulation . . . 71
3.2.3 Numerical Study . . . 78
3.2.4 Conclusions . . . 82
3.3 Frequency Regulation Service Provision in Data Center with Computational Flexibility . . . 82
3.3.1 Overall Framework . . . 82
3.3.2 Hour-Ahead Market Frequency Regulation and Energy Bidding Strategy for Data Center . . . 91
3.3.3 Following Real-Time Frequency Regulation Signal . . . 98
3.3.4 Numerical Study . . . 105
3.3.5 Conclusion . . . 117
3.3.6 Proofs . . . 117

4 Data-driven Volt-VAR Control in Power Distribution Systems **129**
4.1 Related Works . . . 130
4.2 Volt-VAR Control with On-policy Deep Reinforcement Learning Algorithm 132
4.2.1 Problem Formulation . . . 132
4.2.2 Technical Methods . . . 137
4.2.3 Value and Policy Networks . . . 142
4.2.4 Numerical Study . . . 143
4.2.5 Conclusion . . . 148
4.3 Volt-VAR Control with Off-policy Deep Reinforcement Learning Algorithm 148
4.3.1 Problem Formulation . . . 148
4.3.2 Safe Off-policy Deep Reinforcement Learning Algorithm . . . 151
4.3.3 Numerical Study . . . 163
4.3.4 Conclusion . . . 171

5 Algorithmic Trading with Virtual Bids **172**
5.1 Related Works . . . 173
5.2 Portfolio Optimization with Virtual Bids . . . 174
5.2.1 Modeling of Virtual Bids . . . 174
5.2.2 Budget and Risk Constrained Portfolio Optimization . . . 175
5.3 Estimation of Expected Earnings of Virtual Transactions . . . 176
5.3.1 Benchmark Algorithm: An Online Learning Approach . . . 176
5.3.2 A Machine Learning Framework . . . 178
5.4 Risk-Constrained Portfolio Optimization . . . 181
5.5 Numerical Study . . . 183
5.5.1 Profitability of Algorithmic Trading Strategies . . . 184
5.5.2 Influence of Rare Events and Changes in Market Rules . . . 184
5.6 Conclusion and Future Works . . . 186

6 Conclusions **187**

Bibliography **190**

List of Figures

1.1 Residential solar generation capacity in the U.S. 3
1.2 Overview of wholesale electricity market 5

2.1 Numerical error versus load bus voltage 34
2.2 Feasible power injection region of a two-node network with the same supply offer prices on three phases 43
2.3 Feasible power injection region of a two-node network with different supply offer prices on three phases 44

3.1 Overall coordination framework 69
3.2 Electrical wiring diagram of a data center. 71
3.3 Overall framework of frequency regulation service provision by a data center 83
3.4 Prices for frequency regulation services and energy in PJM market 85
3.5 C6 sleep time versus request per second rate 88
3.6 Fitted power consumption curves with default sleep policy 90
3.7 90% pass time versus utilization rate 94
3.8 Request traces of English, Spanish and Polish pages 107
3.9 Prediction of English, Spanish and Polish page visits 107
3.10 Hourly-averaged request arrival rate after scaling 111
3.11 Frequency regulation signal following of one hour for the three different pages 111
3.12 Distribution of requests response time 114
3.13 Frequency regulation service bidding capacity versus predicted price difference with weekly risk limit of $336 115
3.14 Weekly extra net earnings versus the risk limit 116
3.15 Dynamic power consumption per request 123

4.1 Structure of the policy network 143
4.2 Comparison of voltage profiles on the 4-bus test feeder 146
4.3 Device-decoupled structure of the policy neural network 162
4.4 Average weekly return and voltage violation for 4-bus test feeder 167
4.5 Average weekly return and voltage violation for 34-bus test feeder 168
4.6 Average weekly return and voltage violation for 123-bus test feeder 168

5.1 Histogram of price spreads of a sample node in year 2015 180
5.2 Cumulative net gains of the proposed and the benchmark trading strategies from 2016 to 2017 with $100k daily budget and risk limit 185

List of Tables

2.1 Comparison between three-phase DCOPF and three-phase ACOPF 33
2.2 Three-phase LMPs with only energy and loss components 35
2.3 Three-phase LMPs with energy, loss, and phase imbalance components . . . 35
2.4 Comparison of traditional methods and the convex iteration method with different prices for DERs 57
2.5 Comparison of the SDP relaxation method and the convex iteration method with different prices for three phases 58
2.6 Comparison of the SDP relaxation method and the convex iteration method with same prices for three phases 59
2.7 Comparison of the penalized SDP method and the convex iteration method 60
2.8 Computation time of IEEE 4-bus test feeder 61
2.9 Computation time of IEEE 13-bus test feeder 62
2.10 Scalability of proposed algorithm 63

3.1 Fixed and flexible load profile 79
3.2 LMPs and data center electricity consumption 80
3.3 Electricity cost and total surpluses 81
3.4 Forecast performance comparison among English, Spanish, and Polish web pages 106
3.5 Extracted features 108
3.6 Performance of the price difference forecast 109
3.7 Frequency regulation signal following performance scores 113
3.8 Energy costs with/without providing frequency regulation service 114

4.1 Performance comparison of Volt-VAR control algorithms 146
4.2 Computation time of Volt-VAR control algorithms 147
4.3 Parameter settings for reinforcement learning algorithms 165
4.4 Performance comparison of Volt-VAR control algorithms 169
4.5 Comparison between CSAC and SAC for Volt-VAR control 170

5.1 Performance comparison of algorithmic trading strategies 185

Chapter 1

Introduction

Optimization techniques are used in various applications for the operation and management of electric power grids. For example, the classical optimization problems include the unit commitment problem at a transmission network level that determines the set of generators to be turned on and off, the economic dispatch problem that settles the generation dispatches and locational marginal prices, and the Volt-VAR control problem that adjusts the tap positions of voltage regulating devices in distribution systems. Due to the integration of renewable energy resources, the installation of energy storage devices, and the construction of data centers, we are facing new challenges and optimization problems in the operation of power grids. Fortunately, the advanced communication and sensing technologies are gradually deployed in the smart grid, which enables more flexible and complex control. It is the right time and also a critical time to develop new optimization and control solutions to realize a more efficient, stable and reliable power grid.

In this chapter, the background and the current status of smart grids will be

introduced first in Section 1.1. The overview of the technical challenges and research opportunities are described in Section 1.2. The contribution of this dissertation is summarized in Section 1.3. Lastly, the organization of the dissertation is provided in Section 1.4.

1.1 Background

Our electricity grids have been transformed significantly in various aspects including generation, demand, and control over the last few decades. The renewable generation gradually replaces the fuel generation in order to reduce the greenhouse emission. As the electrification and digitization of people's daily life continue, new types of load like electric vehicle charging and computing services arise. Moreover, with the advancement of communication technologies, data-driven control methods start to attract more and more interests for the automation of power grids. In this chapter, the current situation and status of smart grids are described.

1.1.1 Increasing Penetration of Distributed Energy Resources

The penetration of distributed energy resources, including distributed solar and wind generations, energy storage facilities, is growing rapidly. For example, the total residential solar generation capacity increases from 3346 MW in year 2014 to 13526 MW in year 2019 according to the EIA-861M dataset [1]. Moreover, the capacity of battery storage in the U.S. was only 862 MW at the end of year 2018 but projected to be over 2,500 MW by 2023 [2]. Benefiting from the new battery technology, the sales of plug-in electrical vehicles (EVs) also soared from 118k in 2014 to 358k in 2018 [3]. The large amount of charging

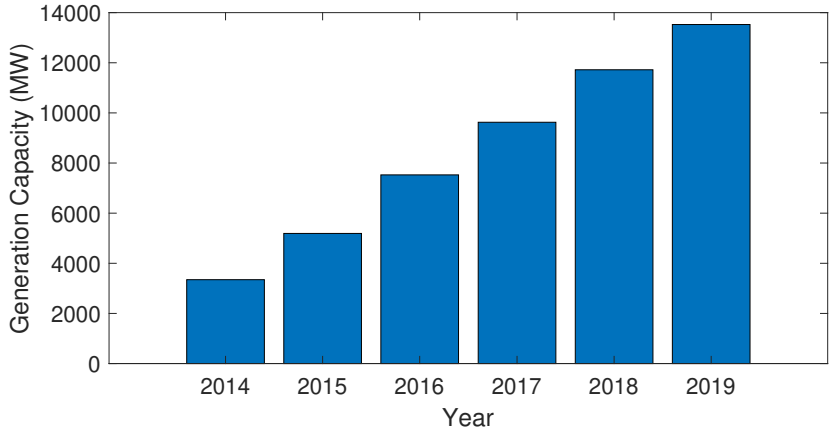

Figure 1.1: Residential solar generation capacity in the U.S.

demand from EVs and energy storage devices challenges the stability of the distribution grids. However, they can be better coordinated along with the renewable energy resources to increase overall operation efficiency.

1.1.2 Emergent Computing Load

Due to the explosive growth of digital data, cloud computing services, including servers, storage, software, and data analytics over the cloud, are becoming more and more popular for the scalability and flexibility. The emergence of cloud computing services drove the rapid expansion of data centers. The number of hyper-scale data centers was over 500 in Q3 of year 2019, which was about 16% increase from 430 at the end of 2018 [4].

With continuous new construction, data centers have become a significant energy consumer in the U.S.. According to a recent U.S. data center energy usage report [5], around 70 TWh of electricity was consumed by data centers in 2014 and the annual shipment of data center servers is expected to grow 3% annually through 2020. The operational cost is

a major component of the total cost of ownership of a data center. The electricity cost is about 30% to 50% of the total operational cost [6]. From the view of data center operators, it is imperative to improve energy management strategies and lower energy bills.

1.1.3 Widespread Deployment of Smart Meters and Sensors

The advanced metering infrastructure, i.e., smart meters, has been widely deployed in the U.S., which enables instantaneous two-way communication between utility companies and customers. In 2018, about 86.8 million smart meters had been installed by U.S. utility companies [7]. Moreover, with gradually reduced manufacturing cost, the smart sensors have been installed for both overhead and underground monitoring.

The physical system model based control is typically adopted by traditional distribution management system. However, the robust parameter estimation is not always available. With high resolution data from the smart meters and sensors, the data-driven based control could be the future of distribution automation.

1.1.4 Wholesale Electricity Market

Wholesale electricity market in the U.S. have been established for decades to provide reliable electricity services with the minimum cost at a transmission system level. The main products in the wholesale markets are energy, capacity and ancillary services. There are four key players that participate in the market: system operator, supplier, consumer and financial trader. The suppliers produce products and sell them into the market. The consumers buy products from the markets and consume them. The financial traders can

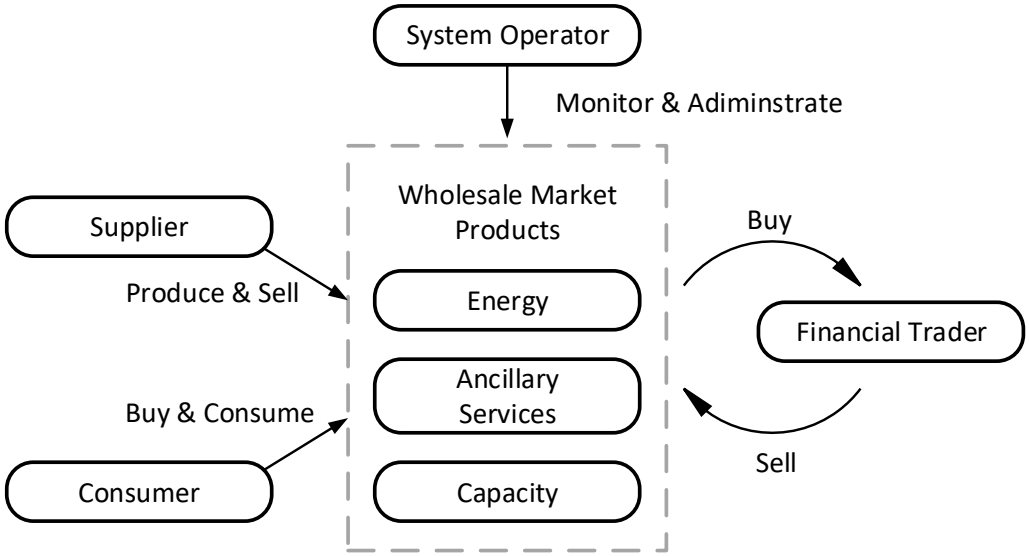

Figure 1.2: Overview of wholesale electricity market

also participate the market through financial tools like virtual bids without owning any physical assets. The system operators collect the offers and bids from the other participants, settle the electricity prices and dispatch points, and maintain the stable operation of transmission networks.

1.2 Technical Challenges and Research Opportunities

With continuous integration of distributed energy resources (DERs) into power grids, new opportunities and challenges arise at the same time. From the operators' perspective, it is critical to better coordinate the DERs in distribution systems to reduce the total energy cost, mitigate the phase unbalance, and improve the power quality. From the consumers' perspective, with the ability to adjust their consumption, they can participate

in the ancillary service or demand response program to further reduce the total energy bills. Moreover, with advancing metering, sensing and digital management system, the reliability and efficiency of power grids could be further improved with data driven methods. In this section, the overview of the four targeted problems in this dissertation is provided.

1.2.1 Three-phase OPF in Distribution Systems

A two-stage pricing approach for residential demand response management is proposed recently [8]. Under the two-stage pricing framework, an innovative proactive demand participation scheme is proposed with demand bid curve forecasting [9]. In the two-stage pricing framework, the market-based dispatch is extended from wholesale markets to retail markets at a distribution network level, where a three-phase optimal power flow (OPF) problem needs to be solved.

Because of the balanced three-phase loadings in transmission systems, low resistance over reactance ratio of transmission lines, a single-phase direct current optimal power flow (DCOPF) or alternating current optimal power (ACOPF) problem is typically solved for the economic dispatch in wholesale markets. The single-phase DCOPF, as a linear programming (LP) problem, can be solved quickly and robustly. The single-phase ACOPF problem is non-linear and non-convex. It is more difficult to be solved but achieves more accurate results.

However,with unbalanced loadings among three phases and much larger resistance of distribution lines, the single-phase OPF formulation is no longer suitable for the distribution system. The three-phase OPF problems need to be solved instead in order to achieve better dispatch results. To have a better management of distribution networks, it

is necessary to develop new algorithms based on the updated problem formulations.

1.2.2 Ancillary Services Provision with Data Centers

In distribution systems, ancillary services are needed to mitigate the unbalanced three-phase loadings issues. High degree of unbalance in distribution feeders can significantly affect power quality, damage electrical equipment and appliances [10], and result in highly unbalanced three-phase voltages. In addition, unbalanced systems are more likely to experience overloading on a phase wire or a neutral wire. The overloading will not only cause overheating but also lead to tripping of a protective device if there is large neutral current [11].

In transmission systems, the intermittency of renewable generation outputs poses a new set of operational and planning challenges to power system operators. In particular, there is an increasing need for high quality frequency regulation services to balance the supply and demand of electricity in real-time and mitigate the uncertainties in renewable generation outputs [12]. A major challenge of automatic generation control (AGC) in fossil-fueled power plants is that they are not well suited to follow the AGC set points on a second-by-second basis with very high accuracy.

Data centers could provide phase balancing services in distribution systems and frequency regulation services in transmission systems by adjusting their power consumption with different techniques including dynamic voltage and frequency scaling (DVFS) [13, 14, 15], virtual machine migration and auto-scaling [16, 17], and geometrical load balancing [18]. By providing ancillary services, data centers can not only receive compensations and reduce energy bills but also help further improve the stability and reliability of power grids.

1.2.3 Data-driven Volt-VAR Control

As the penetration level of distributed energy resources (DERs) continuously increases in power distribution systems, it is increasingly difficult to keep the voltages along the feeders within desired ranges. The voltage profile highly impacts the electricity service quality for end users. Both over-voltage and under-voltage conditions could reduce energy efficiency, cause equipment malfunction, and damage customers' electrical appliances. Equipped with remotely controlled and monitoring devices, electric utilities started adopting Volt-VAR control (VVC) to maintain voltages within allowable ranges, manage power factor, and reduce operation costs. These control objectives can be achieved by coordinating the operations of various equipment such as voltage regulators, on-load tap changers, switchable capacitor banks, and smart inverters.

Although successful field demonstrations of VVC have been reported by many electric utilities, there are still many barriers to the wide-spread adoption of the technology. One of the most significant barriers is the lack of robust distribution network topology and parameter information, which are required in optimization based VVC approaches. In particular, inaccurate distribution secondary systems' information [19, 20, 21] makes it difficult for VVC to ensure that customers' voltages will stay within the acceptable range. Moreover, the model-based control approaches are not always scalable and may not be applicable in real-time control environments.

With gradually reduced manufacturing cost, the smart meters, sensors, and remotely controlled devices are widely installed in power grids. At the same time, the communication infrastructure is also upgrading, which enables instantaneous communication

between operators and devices. All these new technologies make the data-driven control a promising direction to overcome the barriers of traditional approaches.

1.2.4 Algorithmic Trading with Virtual Bids

The U.S. energy market is designed as a two-settlement market, which consists of the day-ahead (DA) market and the real-time (RT) market. The DA market is for day-ahead planning with security constrained unit commitment and security constrained economic dispatch programs that simultaneously optimize energy and reserves. The RT market is settled based on the actual system operation conditions.

Financial traders without physical assets can buy or sell energy in the DA market with an explicit requirement to sell or buy it back in the RT market through virtual bids. The virtual bids are introduced into the two-settlement electricity market to improve market efficiency, promote price convergence [22], provide hedging instruments [23], and enhance market liquidity.

For proprietary trading firms, it is imperative to design virtual bids portfolio trading strategies that maximize the expected earnings and minimize risk. Moreover, the trading profitability of virtual bids can be used as an important metric to evaluate the relative market efficiency across different whole electricity markets.

1.3 Contributions

This dissertation primarily studies the four targeted problems mentioned in the previous section. The main contributions of this dissertation include:

Three-phase DCOPF and three-phase ACOPF: For the three-phase OPF problems in distribution systems, we first fill the knowledge gap by extending the iterative single-phase DCOPF algorithm to three-phase system with fictitious nodal demand (FND) [24]. The proposed iterative three-phase DCOPF algorithm provides not only a computationally efficient solution but also a good approximation to the ACOPF solutions. In addition, none of the existing literatures have touched on the subject of LMP decomposition in three-phase distribution systems. We presents a generalized three-phase LMP decomposition within the DCOPF framework.

Furthermore, we develop a computationally efficient and scalable three-phase OPF algorithm which is capable of finding optimal solutions for the three-phase ACOPF problem [25]. Specifically, we first revisit the rank conundrum in solving three-phase OPF problems. The innovative three-phase OPF algorithm synergistically combines the convex iteration technique and the chordal based conversion algorithm. We also propose a greedy algorithm to find an appropriate grid partitioning scheme that results in lower computational complexity.

Phase balancing and frequency regulation service provisioning with data centers: For distribution systems, we propose to solve the network phase balancing problem by shifting computational loads among the servers connected to three different phase wires in a data center [26]. An iterative scheme to coordinate the operations of data center and distributed energy resources within a DSO managed electricity market is proposed. We also derived the three-phase Locational Marginal Prices (LMPs) sensitivities in a distribution market and embedded the price sensitivities into the data center's electricity cost

minimization problem. The operational coordination strategy for data center and DERs is very effective in reducing phase unbalance; therefore, improving distribution network operational efficiency and reliability. The simulation results show that the degree of unbalance of a distribution feeder can be reduced by up to 100% and the electricity cost of the data center decreases by more than 4.0%.

At a transmission system level, we design a comprehensive framework for a data center to provide frequency regulation service, consisting of bidding into the hour-ahead electricity market and following frequency regulation signal in real-time operations. A risk-constrained hour-ahead bidding strategy considering uncertainties of energy and frequency regulation service prices is developed to determine the optimal energy and frequency regulation bids by data centers. The dummy computing load is introduced for the first time to increase the amount of frequency regulation service provision of the data center in addition to the DVFS technique. A realistic bi-linear server power consumption model and rule-based data center power consumption control algorithm not only enable accurate frequency regulation signal following but also limit degradation in quality of service (QoS). The theoretical derivation and simulation results point out that the profitability of frequency regulation service provision by data center depends on an accurate prediction of the price difference between frequency regulation service and energy. The simulation results show that for a period of 3 months the proposed frequency regulation service provision framework reduces the electric bill by $\$21,590(8.1\%)$ for a data center with 100,000 servers compared to the power minimization strategy.

Safe deep reinforcement learning algorithms in the application of Volt-VAR

control: We first adopt the constraint policy optimization (CPO)[27] algorithm to solve the Volt-VAR control problem [28], which statistically guarantees every control policy during learning will satisfy operational constraints in the form of expectation. Compared to the optimization-based approaches, our proposed algorithm has better scalability and does not require an accurate and complete physical model of the distribution network.

Furthermore, a safe off-policy deep reinforcement learning (DRL) algorithm is developed [29]. Instead of penalizing constraints violation in the reward function of markov decison process (MDP), we propose a constrained MDP (CMDP) formulation for the VVC problem, which explicitly models the physical operation constraints. By synergistically combining the merits of the method of multipliers and soft actor-critic (SAC) [30] algorithm, our proposed constraint soft-actor critic (CSAC) algorithm can better satisfy the operation constraints in power distribution systems. Compared to tabular Q-learning and deep Q-network (DQN) [31, 32], our proposed CSAC algorithm has significantly improved scalability. By designing the policy network with a device-decoupled structure, the number of parameters only increases linearly with the number of voltage regulating devices. On the other hand, in the Q-learning based approaches, the number of network parameters increases exponentially with the number of voltage regulating devices. As an off-policy method, the proposed algorithm is more sample efficient than the state-of-the-art DRL algorithms for CMDP such as constrained policy optimization [27]. This is because our proposed method can effectively reuse historical operational data for training purpose. Furthermore, by using an ordinal network structure to encode the natural ordering between discrete actions of voltage regulating devices, the inductive bias can be introduced to further accelerate the

learning process.

Algorithmic trading with virtual bids in electricity markets: A machine learning based trading strategy is developed [33] for financial traders to maximize total profits and limit portfolio risks. We push the research frontier of algorithmic trading with virtual bids in the following ways. First, instead of relying only on historical LMPs to model and estimate the DA and RT price spreads and payoff of virtual bids, this paper develops a mixture density network (MDN) to infer the conditional distribution of nodal price spreads given the fundamental inputs such as electric load, generation outage, and transmission outage. Second, a risk-constrained portfolio optimization problem for virtual bids is formulated and efficiently solved with a finite number of scenarios. Third, a machine learning framework for algorithmic trading with virtual bids is established by synergistically combining the risk-constrained portfolio optimization framework and the MDN model. The profitability of our proposed trading strategy outperforms the state-of-the-art online learning (OL) approach to virtual trading [34].

1.4 Organization

The remainder of the dissertation is organized as follows. In Chapter 2, the three-phase OPF problems in distribution network are investigated, including the three-phase DCOPF [24] problem and three-phase ACOPF problem [25]. In Chapter 3, data centers are used as flexible resources to provide ancillary services in power grids, including the phase balancing service in distribution networks [26] and frequency regulation service in transmission networks [35]. The Volt-VAR control with deep reinforcement learning algo-

rithms including both on-policy [28] and off-policy [29] algorithms are proposed in Chapter 4. A machine learning framework for algorithmic trading with virtual bids [33] is developed in Chapter 5. Chapter 6 states the conclusions.

Chapter 2

Three-phase OPF in Distribution System

The increasing penetration of DERs has significantly influenced the daily operation of distribution systems. It is critical to manage the DERs in an efficient and effective way. A two-stage pricing approach for residential demand response management is proposed in [8] recently, where a economic dispatch problem need to be solved at a distribution system level. Because of unbalanced loads and relative larger conductance of distribution lines, the three-phase OPF models are more appropriate for distribution systems. In this section, we investigate the three-phase DCOPF and ACOPF problems in distribution systems.

2.1 Related Works

Many researchers have studied the problem of DERs coordination and management. In [36], the concept of LMP for distribution system is first proposed in order to

manage distribution generation (DG) resources and reduce lines losses. A real-time pricing strategy is used to schedule load with a linear-programming (LP) method in [37]. Researchers in [8] provide a two-stage pricing approach for residential demand response management. In [38, 9], an innovative proactive demand participation scheme is proposed under two-stage pricing framework with demand bid curve forecasting. To mitigate power quality issues of micro-grid, a mixed integer programming (MIP) approach is studied in [39]. The OPF problem is one of the core problems to coordinate the DERs.

The single-phase OPF problem for the transmission system has been studied extensively in the past 50 years. The transmission system can be treated as a single-phase system in the OPF problem due to the relatively balanced electricity loads across three phases and periodically transposed transmission lines. The single-phase OPF problem is highly non-convex due to the nonlinear relationship between voltage and power injections. This problem can be solved by numerous algorithms including Newton-based methods [40, 41], linear and quadratic programming [42], nonlinear and polynomial programming [43], interior point methods [44], and heuristic optimization methods [45]. However, none of them guarantees a global optimum. To obtain a global optimum, a SDP relaxation method was recently proposed [46]. The method first transforms the OPF problem to a semidefinite programming problem (SDP) where the only non-convex constraint is a rank-one constraint. If the rank-one constraint is dropped, then convex optimization techniques can be used to solve the problem. The global optimality of this convex relaxation method has been proven for single-phase tree-networks [47] and a small group of mesh networks [48] with some small perturbations in the admittance matrix. Nonetheless, a rank-one solution can not always

be achieved with the convex relaxation algorithm. The exactness of the convex relaxation has been investigated in [49, 50]. The convex relaxation approach has been leveraged to develop heuristic algorithms that solve rank-constrained optimization problems [51, 52, 53]. However, the convergence of these algorithms cannot be guaranteed.

Only a few researchers have studied the three-phase OPF problem [54, 55]. A quasi-Newton method based approach was developed after transforming the OPF problem with implicit function theorem in [54]. Authors in [55] developed a distributed semidefinite programming solver for the three-phase OPF problem based on ADMM and the Lagrangian relaxation method. However, neither of the algorithms guarantees convergence or global optimality. Furthermore, these algorithms are not computationally efficient enough to handle realistic distribution feeders with thousands of buildings and customers.

In Section 2.2, we first fill the knowledge gap by extending the iterative single-phase DCOPF algorithm [56] to three-phase system with fictitious nodal demand (FND) [24]. Secondly, we develop a computationally efficient and scalable for the non-linear non-convex three-phase OPF algorithm [25] in Section 2.3.

2.2 Three-phase DCOPF

2.2.1 Problem Formulation

List of Symbols

B_{ik}^{pm}, G_{ik}^{pm} Susceptance and conductance between node i with phase p and node k with phase m.

$Cd\,(n,m)_j$ Demand bid price of the j-th segment of price sensitive demand bid curve at node n with phase m.

$Cg\,(1)_i$ Supply offer price of the i-th segment of supply offer curve at reference bus.

$Cg\,(n,m)_i$ Supply offer price of the i-th segment of supply offer curve at node n with phase m.

$d\,(n,m)_j$ Demand bid quantity of the j-th segment of price sensitive demand bid curve at node n with phase m.

$(DF_s^t)^p$ Phase p's delivery factor at node s with phase t.

FD_n^m Real power of fixed demand at node n with phase m.

FP, FQ Set of real and reactive power branch flows.

FP_b^p, FQ_b^p Real and reactive power flow on branch b with phase p.

$g\,(n,m)_i$ Supply offer quantity of the i-th segment of supply offer curve at node n with phase m.

$GSFP_{ik\_q}^{p\_g}$ Generation shift factor for real power flow of the branch which connects node i and k with phase p when power injection is at node q with phase g.

$GSFQ_{ik\_q}^{p\_g}$ Generation shift factor for reactive power flow of the branch which connects node i and k with phase p when power injection is at node q with phase g.

J_1 Total number of segments of demand bid curve at the reference bus.

J_n^m, K_n^m	Total number of segments of supply offer curve and demand bid curve at node n with phase m.
I_{ik}, V_{ik}	Current and voltage across the branch connecting node i and k.
$(LF_s^t)^p$	Phase p's loss factor at node s with phase t.
N	Total number of nodes including the swing bus.
PD_n^m	Real power of total demand at node n with phase m.
PG_n^m	Real power of generation at node n with phase m.
P_i^p, Q_i^p	Net injection of real and reactive power injection at node i with phase p.
P_{ik}^p, Q_{ik}^p	Real and reactive power flowing from node i to node k with phase m.
P_{loss}^p	Total real power losses at phase p.
$PLimit_{ik}^p$	Real power flow limit between node i and k with phase p.
R_{ik}^{pg}, X_{ik}^{pg}	Resistance and reactance of the phase impedance matrix relating node i with phase p and node k with phase g.
S_{ik}^p	Complex power flowing from node i to node k with phase m.
$(S_{loss})_{ik}$	Complex power losses of the branches connecting node i and k.
$SLimit_{ik}^p$	Complex power flow limit between node i and k with phase p.
Z_{ik}	Phase impedance matrix of the line connecting node i and k.
γ	Power imbalance limit between phases.

$\theta_{V_{ik}^{pm}}$,$\theta_{I_{ik}^{pg}}$ Voltage angle difference and current angle difference between node i with phase p and node k with phase m.

A. Linear Model without Considering Loss

The objective of three-phase DCOPF problem is to maximize total surplus of customers and producers in a distribution system. On the supply side, an equivalent system supply offer curve is created at the point-of-integration to the transmission system. On the demand side, individual buildings and customers express their energy usage preferences by constructing price-sensitive demand bid curves [38]. Node 1, the point-of-integration to the transmission system, is selected as the swing bus of the distribution system. Note that there is only one supply offer curve for all three phases at the distribution substation. The objective function of the DCOPF problem is provided in equation (2.1). Without considering losses, the real power balance constraints are represented by equation (2.2). In subsection 2.2.1, these constraints are modified when real power losses are taken into consideration. Equation (2.3) shows the power flow limit constraints. Generating shift factors used in the equation are derived in subsection 2.2.1. Phase imbalance constraints are represented in equation (2.4), which have been shown to be effective in mitigating phase imbalance problems in [3].

$$\max_{d} \sum_{n=2}^{N} \sum_{m=1}^{3} \left(\sum_{j=1}^{K_n^m} C_d\left(n,m\right)_j d\left(n,m\right)_j - \sum_{i=1}^{J_n^m} Cg\left(n,m\right)_i g\left(n,m\right)_i \right) - \sum_{i=1}^{J_1} Cg\left(1\right)_i g\left(1\right)_i \quad (2.1)$$

subject to:

$$\sum_{n=1}^{N} PG_n^m = \sum_{n=1}^{N} PD_n^m, m = 1,2,3 \quad (2.2)$$

$$|\sum_{q=2}^{N}\sum_{g=1}^{3} GSFP_{ik\_q}^{p\_g}\cdot\left(PG_q^g - PD_q^g\right)| \le PLimit_{ik}^{p}, \forall i,k \quad and \quad i\neq k \tag{2.3}$$

$$|\sum_{n=2}^{N} P_n^i - \sum_{n=2}^{N} P_n^j| \le \gamma,\ i,j=1,2,3\ \ and\ \ i\neq j \tag{2.4}$$

where

$$PLimit_{ik}^{p} = \sqrt{(SLimit_{ik}^{p})^2 - (\sum_{q=2}^{N}\sum_{g=1}^{3} GSFQ_{ik\_q}^{p\_g}\cdot Q_q^g)^2} \tag{2.5}$$

B. Derivation of Generation Shift Factors

The relationship between real power injection and voltage angle is derived by differentiating load flow equation with respect to θ_V. We start the derivation from equations (2.6)-(2.7) [57]:

$$\frac{\partial P_i^p}{\partial {\theta_V}_k^m} = |V_i^p||V_k^m|[G_{ik}^{pm} sin{\theta_V}_{ik}^{pm} - B_{ik}^{pm} cos{\theta_V}_{ik}^{pm}],\ p\neq m \text{ or } i\neq k \tag{2.6}$$

$$\frac{\partial P_i^p}{\partial {\theta_V}_i^p} = -B_{ii}^{pp}(V_i^p)^2 - Q_i^p \tag{2.7}$$

Under most operational scenarios, the voltage drop and voltage angle bias are small when the distribution network is not heavily loaded or seriously unbalanced. When large voltage drop happens, step-type voltage regulators, load tap changing transformers, and shunt capacitors will be operated to keep customers' voltage within an acceptable range. Thus the following assumptions are made:

$$|V_i^p| \approx 1 \tag{2.8}$$

$${\theta_V}_{ik}^{pm} \approx \begin{cases} 120^\circ & \text{if } p-m=-1,2 \\ -120^\circ & \text{if } p-m=1,-2 \\ 0^\circ & \text{if } p-m=0 \end{cases} \tag{2.9}$$

With the above assumptions, equations (2.6) and (2.7) can be simplified as:

$$\frac{\partial P_i^p}{\partial {\theta_V}_k^m} = G_{ik}^{pm} sin{\theta_V}_{ik}^{pm} - B_{ik}^{pm} cos{\theta_V}_{ik}^{pm}, p \neq m \text{ or } i \neq k \tag{2.10}$$

$$\frac{\partial P_i^p}{\partial {\theta_V}_i^p} = -B_{ii}^{pp} - Q_i^p \tag{2.11}$$

Excluding the swing bus, in condensed form equations (2.10)-(2.11) become:

$$\Delta P = [B_P] \Delta \theta_V \tag{2.12}$$

where $[B_P]$ is a $3\,(N-1) \times 3\,(N-1)$ matrix.

The relationship between reactive power injection and voltage magnitude is derived by differentiating load flow equation with respect to V[57]:

$$\frac{\partial Q_i^p}{\partial V_k^m} = |V_i^p|[G_{ik}^{pm} sin{\theta_V}_{ik}^{pm} - B_{ik}^{pm} cos{\theta_V}_{ik}^{pm}], p \neq m \text{ or } i \neq k \tag{2.13}$$

$$\frac{\partial Q_i^p}{\partial V_i^p} = \sum_{k=1}^{N} \sum_{m=1}^{3} |V_k^m|[G_{ik}^{pm} sin{\theta_V}_{ik}^{pm} - B_{ik}^{pm} cos{\theta_V}_{ik}^{pm}] - |V_i^p| B_{ii}^{pp} cos{\theta_V}_{ii}^{pp} \tag{2.14}$$

With the same assumption above, equation (2.13) can be simplified as:

$$\frac{\partial Q_i^p}{\partial V_k^m} = G_{ik}^{pm} sin{\theta_V}_{ik}^{pm} - B_{ik}^{pm} cos{\theta_V}_{ik}^{pm}, p \neq m \text{ or } i \neq k \tag{2.15}$$

As the shunt component is usually very small

$$\sum_{k=1}^{N} B_{ik}^{pm} \approx 0, \quad \sum_{k=1}^{N} G_{ik}^{pm} \approx 0, \quad m = 1, 2, 3$$

Thus

$$\frac{\partial Q_i^p}{\partial V_i^p} \approx -B_{ii}^{pp} \tag{2.16}$$

Excluding the swing bus, in condensed form equations (13)-(14) become:

$$\Delta Q = [B_Q] \Delta V \tag{2.17}$$

where $[B_Q]$ is a $3\,(N-1) \times 3\,(N-1)$ matrix.

The complex power flowing from node i to k with phase p is given by (2.18):

$$S_{ik}^{p} = V_i^{p} \sum_{m=1}^{3} [\left(G_{ik}^{pm} + jB_{ik}^{pm}\right)\left(V_i^{m} - V_k^{m}\right)]^{*} \tag{2.18}$$

By separating the real and imaginary part of complex branch flow equation (17), we get equations (2.19) and (2.20).

$$\begin{aligned} P_{ik}^{p} = \sum_{m=1}^{3} \{ & |V_i^{p}||V_i^{m}| cos\theta_{V_{ii}}^{pm} G_{ik}^{pm} - |V_i^{p}||V_k^{m}| cos\theta_{V_{ik}}^{pm} G_{ik}^{pm} \\ & + |V_i^{p}||V_i^{m}| sin\theta_{V_{ii}}^{pm} B_{ik}^{pm} - |V_i^{p}||V_k^{m}| sin\theta_{V_{ik}}^{pm} B_{ik}^{pm} \} \end{aligned} \tag{2.19}$$

$$\begin{aligned} Q_{ik}^{p} = \sum_{m=1}^{3} \{ & |V_i^{p}||V_i^{m}| sin\theta_{V_{ii}}^{pm} G_{ik}^{pm} - |V_i^{p}||V_k^{m}| sin\theta_{V_{ik}}^{pm} G_{ik}^{pm} \\ & - |V_i^{p}||V_i^{m}| cos\theta_{V_{ii}}^{pm} B_{ik}^{pm} + |V_i^{p}||V_k^{m}| cos\theta_{V_{ik}}^{pm} B_{ik}^{pm} \} \end{aligned} \tag{2.20}$$

Equation (2.19) can be simplified as follows by the assuming $|V_i^{p}| \approx 1$.

$$\begin{aligned} P_{ik}^{p} = & \sum_{m=1}^{3} 2B_{ik}^{pm} sin\left(\frac{\theta_{V_i}^{p} - \theta_{V_i}^{m} - \theta_{V_i}^{p} + \theta_{V_k}^{m}}{2}\right) cos\left(\frac{\theta_{V_i}^{p} - \theta_{V_i}^{m} + \theta_{V_i}^{p} - \theta_{V_k}^{m}}{2}\right) \\ & - \sum_{m=1}^{3} 2G_{ik}^{pm} sin\left(\frac{\theta_{V_i}^{p} - \theta_{V_i}^{m} - \theta_{V_i}^{p} + \theta_{V_k}^{m}}{2}\right) sin(\frac{\theta_{V_i}^{p} - \theta_{V_i}^{m} + \theta_{V_i}^{p} - \theta_{V_k}^{m}}{2}) \end{aligned} \tag{2.21}$$

If we assume balanced voltage angles,

$$\theta_{V_i}^{p} - \theta_{V_i}^{m} + \theta_{V_i}^{p} - \theta_{V_k}^{m} \approx \begin{cases} 240^{\circ} & \text{if } p-m = -1, 2 \\ -240^{\circ} & \text{if } p-m = 1, -2 \\ 0^{\circ} & \text{if } p-m = 0 \end{cases} \tag{2.22}$$

We have $sin\theta_{V_{ik}}^{mm} \approx \theta_{V_{ik}}^{mm}$. Now equation (2.21) can be simplified as follows.

$$P_{ik}^{p} = \sum_{m=1}^{3} \left(B_{P_{ik}}^{pm}\right)^{"} \left(\theta_{V_i}^{m} - \theta_{V_k}^{m}\right) \tag{2.23}$$

where

$$\left(B_{P_{ik}}^{pm}\right)^{"} = \begin{cases} \frac{1}{2}B_{ik}^{pm} + \frac{\sqrt{3}}{2}G_{ik}^{pm} & \text{if } p-m = -1, 2; \\ \frac{1}{2}B_{ik}^{pm} - \frac{\sqrt{3}}{2}G_{ik}^{pm} & \text{if } p-m = 1, -2; \\ -B_{ik}^{pm} & \text{if } p-m = 0. \end{cases} \tag{2.24}$$

According to equation (2.23), the change in real power branch flow ΔP_B can be represented in condensed form as:

$$\Delta P_B = [D_P][A]\Delta\theta_V \tag{2.25}$$

where ΔP_B is a $3L \times 1$ vector. L is the total number of branches. $[D_P]$ is a $3L \times 3L$ matrix, whose off-diagonal 3×3 blocks are zeros. Let D_{Pb} denote the b-th 3×3 diagonal block connecting bus i and bus k

$$D_{Pb} = \begin{bmatrix} -B_{ik}^{11} & \frac{1}{2}B_{ik}^{12} + \frac{\sqrt{3}}{2}G_{ik}^{12} & \frac{1}{2}B_{ik}^{13} - \frac{\sqrt{3}}{2}G_{ik}^{13} \\ \frac{1}{2}B_{ik}^{21} - \frac{\sqrt{3}}{2}G_{ik}^{21} & -B_{ik}^{22} & \frac{1}{2}B_{ik}^{23} + \frac{\sqrt{3}}{2}G_{ik}^{23} \\ \frac{1}{2}B_{ik}^{31} + \frac{\sqrt{3}}{2}G_{ik}^{31} & \frac{1}{2}B_{ik}^{32} - \frac{\sqrt{3}}{2}G_{ik}^{32} & -B_{ik}^{33} \end{bmatrix}$$

$[A]$ is a $3L \times 3(N-1)$ node-arc incidence matrix. $[A]$ is compromised of $L \times (N-1)$, 3 by 3 blocks. Each row of the 3×3 blocks represents a three-phase branch. Each column of the 3×3 blocks represents a bus. Let $[A_{ij}]$ be the ij-th 3×3 block of $[A]$.

$$\text{Diagonals of } A_{ij} = \begin{cases} 1 & \text{if branch } i \text{ starts at node } j \\ -1 & \text{if if branch } i \text{ ends at node } j \\ 0 & \text{otherwise} \end{cases} \tag{2.26}$$

The non-diagonal elements of $[A_{ij}]$ are zeros.

Substituting equation (2.12) into (2.25) yields

$$\begin{aligned} \Delta P_B &= [D_P][A]\Delta\theta_V \\ &= [D_P][A][B_P]^{-1}\Delta P \end{aligned} \tag{2.27}$$

Therefore, three-phase generation shift factor matrix for real power flow is derived as:

$$[GSFP] = [D_P][A][B_P]^{-1} \tag{2.28}$$

With $|V_i^p| \approx 1$ and the balanced angle assumption:

$$\theta_{V}{}_{ii}^{pm} \approx \theta_{V}{}_{ik}^{pm} = \begin{cases} 120^\circ & \text{if } p-m=-1,2; \\ -120^\circ & \text{if } p-m=1,-2? \\ 0^\circ & \text{if } p-m=0. \end{cases} \tag{2.29}$$

Equation (2.20) can be simplified as:

$$Q_{ik}^p = \sum_{m=1}^{3} \left(G_{ik}^{pm} sin\theta_V{}^{pm} - B_{ik}^{pm} cos\theta_V{}^{pm} \right) \left(|V_i^m| - |V_k^m| \right) \tag{2.30}$$

Therefore

$$Q_{ik}^p = \sum_{m=1}^{3} (B_{Q}{}_{ik}^{pm})'' (|V_i^m| - |V_k^m|) \tag{2.31}$$

where

$$\left(B_{Q}{}_{ik}^{pm}\right)'' = \begin{cases} \frac{1}{2}B_{ik}^{pm} + \frac{\sqrt{3}}{2}G_{ik}^{pm} & \text{if } p-m=-1,2; \\ \frac{1}{2}B_{ik}^{pm} - \frac{\sqrt{3}}{2}G_{ik}^{pm} & \text{if } p-m=1,-2; \\ -B_{ik}^{pm} & \text{if } p-m=0. \end{cases} \tag{2.32}$$

According to equation (2.31), the change in reactive power branch flow ΔQ_B can be represented in condensed form as:

$$\Delta Q_B = [D_Q][A]\Delta V \tag{2.33}$$

where Q_B is a $3L \times 1$ vector. L is the total number of branches. D_Q is a $3L \times 3L$ matrix, whose off-diagonal 3×3 blocks are zeros. Let the D_{Q_b} denote the b-th 3×3 diagonal block connecting node i and node k.

$$D_{Q_b} = \begin{bmatrix} -B_{ik}^{11} & \frac{1}{2}B_{ik}^{12} + \frac{\sqrt{3}}{2}G_{ik}^{12} & \frac{1}{2}B_{ik}^{13} - \frac{\sqrt{3}}{2}G_{ik}^{13} \\ \frac{1}{2}B_{ik}^{21} - \frac{\sqrt{3}}{2}G_{ik}^{21} & -B_{ik}^{22} & \frac{1}{2}B_{ik}^{23} + \frac{\sqrt{3}}{2}G_{ik}^{23} \\ \frac{1}{2}B_{ik}^{31} + \frac{\sqrt{3}}{2}G_{ik}^{31} & \frac{1}{2}B_{ik}^{32} - \frac{\sqrt{3}}{2}G_{ik}^{32} & -B_{ik}^{33} \end{bmatrix}$$

Similarly, with equations (2.17) and (2.33), three-phase generation shift factor matrix for reactive power flow is derived as:

$$[GSFQ] = [D_Q][A][B_Q]^{-1} \tag{2.34}$$

The derivations of three-phase GSFs have the same form as single-phase GSF matrix. However, matrices $[D_P]$, $[D_Q]$, $[A]$, $[B_P]$ and $[B_Q]$ are constructed in a different way. Intuitively, the differences arise from the mutual coupling among three phases of distribution system line. All of the non-diagonal elements of $[D]$ in single-phase GSF equation are zeros, while non-diagonal elements of diagonal 3 by 3 blocks of $[D_P]$ and $[D_Q]$ in three-phase GSF equations are typically non-zero. In three-phase equations $[B_P]$ and $[B_Q]$ are constructed with conductance and susceptance from the admittance matrix Y.

C. Centralized Loss Model

The power loss on each branch can be written as:

$$\begin{aligned}(S_{Loss})_{ik} &= V_{ik} \cdot {I_{ik}}^* = (Z_{ik} I_{ik}) \cdot {I_{ik}}^* \\ &= \left(\begin{bmatrix} Z_{ik}^{11} & Z_{ik}^{12} & Z_{ik}^{13} \\ Z_{ik}^{21} & Z_{ik}^{22} & Z_{ik}^{23} \\ Z_{ik}^{31} & Z_{ik}^{32} & Z_{ik}^{33} \end{bmatrix} I_{ik} \right) \cdot {I_{ik}}^*\end{aligned} \tag{2.35}$$

For each phase, we have:

$$(S_{Loss})_{ik}^{p} = {I_{ik}^{p}}^* \sum_{g=1}^{3} Z_{ik}^{pg} I_{ik}^{g} \tag{2.36}$$

where Z_{ik}^{pg} is the element of phase impedance matrix relating node i with phase p and node k with phase g. ${I_{ik}^{p}}^* = |I_{ik}^{p}| e^{-j\theta_{ik}^{p}}$ and $I_{ik}^{g} = |I_{ik}^{g}| e^{j\theta_{ik}^{g}}$

Assume $|I_{ik}^{g}| \approx |I_{ik}^{p}|, for\, p \neq g$, then (2.36) can be simplified as:

$$\begin{aligned}(S_{Loss})_{ik}^{p} &= |I_{ik}^{p}|^2 \sum_{g=1}^{3} Z_{ik}^{pg} \left(cos\theta_{I_{ik}^{pg}} - j \cdot sin\theta_{I_{ik}^{pg}}\right) \\ &= |I_{ik}^{p}|^2 \sum_{g=1}^{3} \left(R_{ik}^{pg} + jX_{ik}^{pg}\right) \left(cos\theta_{I_{ik}^{pg}} - j \cdot sin\theta_{I_{ik}^{pg}}\right)\end{aligned} \tag{2.37}$$

The real part of (2.37) is the real power loss,

$$\begin{aligned}(P_{Loss})_{ik}^{p} &= \sum_{g=1}^{3} |I_{ik}^{p}|^2 \left(R_{ik}^{pg} \cos\theta_{I_{ik}^{pg}} + X_{ik}^{pg} \sin\theta_{I_{ik}^{pg}}\right) \\ &= \sum_{g=1}^{3} \frac{|S_{ik}^{p}|^2}{|V_{i}^{p}|^2} R_{ik}^{pg\prime}\end{aligned} \tag{2.38}$$

where the equivalent resistance obtained from phase impedance matrix relating node i with phase p and node k with phase g is defined as: $R_{ik}^{pg\prime} \triangleq R_{ik}^{pg} \cos\theta_{I_{ik}^{pg}} + X_{ik}^{pg} \sin\theta_{I_{ik}^{pg}}$.

If balanced current angle is assumed,

$$\theta_{I_i}^{p} - \theta_{I_k}^{g} = \begin{cases} 120^\circ & \text{if } p-g = -1, 2 \\ -120^\circ & \text{if } p-g = 1, -2 \\ 0^\circ & \text{if } p-g = 0. \end{cases} \tag{2.39}$$

Then

$$R_{ik}{}' = \begin{bmatrix} R_{ik}^{11} & -\frac{1}{2}R_{ik}^{12} + \frac{\sqrt{3}}{2}X_{ik}^{12} & -\frac{1}{2}R_{ik}^{13} - \frac{\sqrt{3}}{2}{}^{13}_{ik} \\ -\frac{1}{2}R_{ik}^{21} - \frac{\sqrt{3}}{2}X_{ik}^{21} & R_{ik}^{22} & -\frac{1}{2}R_{ik}^{23} + \frac{\sqrt{3}}{2}X_{ik}^{23} \\ -\frac{1}{2}R_{ik}^{31} + \frac{\sqrt{3}}{2}X_{ik}^{31} & -\frac{1}{2}R_{ik}^{32} - \frac{\sqrt{3}}{2}X_{ik}^{32} & R_{ik}^{33} \end{bmatrix} \tag{2.40}$$

If we assume $|V_{i}^{p}| \approx 1$, then (2.38) can be simplified as:

$$(P_{Loss})_{ik}^{p} = \sum_{g=1}^{3} |S_{ik}^{p}|^2 R_{ik}^{pg\prime} = \sum_{g=1}^{3} (|P_{ik}^{p}|^2 + |Q_{ik}^{p}|^2) R_{ik}^{pg\prime} \tag{2.41}$$

Therefore we have

$$P_{Loss}^{p} = P_{Loss}^{p}(FP) + P_{Loss}^{p}(FQ) \tag{2.42}$$

where

$$P^p_{Loss}(FP) = \sum_{b=1}^{B}\sum_{g=1}^{3}\left(FP^p_b\right)^2 R^{pg\prime}_b \tag{2.43}$$

$$P^p_{Loss}(FQ) = \sum_{b=1}^{B}\sum_{g=1}^{3}\left(FQ^p_b\right)^2 R^{pg\prime}_b \tag{2.44}$$

where B is the total number of branches. FP^p_b and FQ^p_b are real and reactive power flow on branch b at phase g. FP and FQ are the set of real and reactive branch flows respectively. F^p_b can be obtained with GSFs and power injections:

$$FP^p_b = \sum_{q=2}^{N}\sum_{m=1}^{3} GSF^{p\_m}_{b\_q} P^m_q \tag{2.45}$$

$$FQ^p_b = \sum_{q=2}^{N}\sum_{m=1}^{3} GSF^{p\_m}_{b\_q} Q^m_q \tag{2.46}$$

Phase p's marginal loss factor (LF) at bus s with phase t is defined as follows:

$$\begin{aligned}(LF^t_s)^p &\triangleq \frac{\partial P^p_{Loss}}{\partial P^t_s} \\ &= \sum_{b=1}^{B}\sum_{g=1}^{3} 2R^{pg\prime}_b \cdot GSFP^{p\_t}_{b\_s} \sum_{q=2}^{N}\sum_{m=1}^{3} GSFP^{p\_m}_{b\_q} P^m_q\end{aligned} \tag{2.47}$$

Phase p's marginal delivery factor (DF) at bus s with phase t is defined as following:

$$(DF^t_s)^p \triangleq \begin{cases} 1-(LF^t_s)^p, t = p, \\ -(LF^t_s)^p, t \neq p \end{cases} \tag{2.48}$$

Loss factor and delivery factor are keys to deriving marginal loss component of LMP. The definitions of three-phase LF and DF are similar to that of single-phase. However, from Equation (38), we can clearly see that in three-phase distribution systems, power losses of one phase is influenced by net loads of the other phases. Delivery factor $(DF^t_s)^p$ is the amount of power delivered from phase p when the load on node s with phase t increases

by 1KW. When t equals p, DF is the sum of increase of load and power losses due to real power flow on phase p. Otherwise, DF is equal to the increase in power losses due to real power flow on phase p.

With the definitions above, it can be proved that

$$\begin{aligned}
&\sum_{s=1}^{N}\sum_{t=1}^{3}(DF_s^t)^p\cdot\left(PG_s^t-PD_s^t\right)\\
&=\sum_{s=1}^{N}\sum_{t=1}^{3}(DF_s^t)^p P_s^t\\
&=\sum_{s=1}^{N}P_s^p-\sum_{s=1}^{N}\sum_{t=1}^{3}\left[\left(\sum_{b=1}^{B}\sum_{g=1}^{3}2\cdot R_b^{pg\prime}GSFP_{b_s}^{p_t}FP_b^p\right)\cdot P_s^t\right]\\
&=\sum_{s=1}^{N}P_s^p-\sum_{b=1}^{B}\sum_{g=1}^{3}\left(2\cdot R_b^{pg\prime}FP_b^p\sum_{s=1}^{N}\sum_{t=1}^{3}GSFP_{b_s}^{p_t}P_s^t\right)\\
&=\sum_{s=1}^{N}P_s^p-2\sum_{b=1}^{B}\sum_{g=1}^{3}R_b^{pg\prime}\left(FP_b^p\right)^2\\
&=-P_{Loss}^p(FP)+P_{Loss}^p(FQ)
\end{aligned}\tag{2.49}$$

Thus the real power balance constraints become:

$$\begin{aligned}
&\sum_{i=1}^{N}\sum_{m=1}^{3}(DF_i^m)^p\cdot PG_i^m-\sum_{i=1}^{N}\sum_{m=1}^{3}(DF_i^m)^p\cdot\ PD_i^m\\
&+P_{Loss}^p(FP)-P_{Loss}^p(FQ)=0,\ p=1,2,3
\end{aligned}\tag{2.50}$$

D. FND Model

Adopting FND can distribute system losses among distribution lines to eliminate significant mismatch at the reference bus. FND-based DCOPF yields a closer approximation to the results of ACOPF, as shown in [56]. E_i^p, FND at bus i with phase p, is defined as

following:

$$E_i^p = \frac{1}{2}\sum_{b=1}^{B_i}\sum_{g=1}^{3}[\left(FP_b^g\right)^2 + \left(FQ_b^g\right)^2]R_b^{pg\prime} \tag{2.51}$$

where B_i is the number of branches connected to bus i. With FND, the power injection at each node becomes:

$$P_q^m = PG_q^m - PD_q^m - E_q^m \tag{2.52}$$

Using FND, branch flow equation (2.45) can be updated as:

$$FP_b^g = \sum_{q=2}^{N}\sum_{m=1}^{3} GSFP_{b\_q}^{g\_m}\left(PG_q^m - PD_q^m - E_q^m\right) \tag{2.53}$$

Thus, power flow constraints (2.3) are revised as:

$$|\sum_{q=2}^{N}\sum_{g=1}^{3} GSFP_{ik\_q}^{p\text{-}g}\cdot\left(PG_q^g - PD_q^g - E_q^m\right)|^2 \le PLimit_{ik}^p, \forall i,k \quad and \quad i \neq k \tag{2.54}$$

Then the values of LFs, DFs, and power losses are updated with the new power injections and power flows calculated from equations (2.52)-(2.53).

Algorithm 1 Iterative algorithm for three-phase DCOPF

1: Initially set LFs, FNDs and power losses to zeros.

2: Solve linear optimization problem using (2.1), (2.4), (2.50), and (2.54).

3: **repeat**

4: Update the values of FNDs, power losses, LFs and DFs using (2.47), (2.48), (2.51) and (2.53).

5: Solve linear optimization problem using (2.1), (2.4), (2.50), and (2.54).

6: **until** the difference of the load and generation dispatch between the current iteration and previous iteration's result is less than the pre-defined tolerance

E. Iterative DCOPF algorithm

The FND-based DCOPF problem is solved iteratively. The iterative algorithm we propose can be briefly described as in Algorithm 1.

F. Three-Phase LMP Decomposition

The Lagrange function of the OPF problem $\mathcal{L}$ is derived from objective function (2.1) and constraints (2.50), (2.4), and (2.54).

$$\begin{aligned}
\mathcal{L} = & \left(\sum_{n=1}^{N} \sum_{m=1}^{3} \left(\sum_{j=1}^{J_n^m} Cd\left(n,m\right)_j d\left(n,m\right)_j - \sum_{i=1}^{I_n^m} Cg\left(n,m\right)_i g\left(n,m\right)_i \right) \right) \\
& - \sum_{p=1}^{3} \lambda^p \left(\sum_{i=1}^{N} \sum_{m=1}^{3} (DF_i^m)^p \cdot PG_i^m \right. \\
& \left. - \sum_{i=1}^{N} \sum_{m=1}^{3} (DF_i^m)^p \cdot (PD_i^m + E_i^m) + P_{Loss}^p(FP) - P_{Loss}^p(FQ) \right) \\
& - \sum_{b=1}^{B} \sum_{p=1}^{3} \mu_b^{p+} \left(\sum_{q=1}^{N} \sum_{g=1}^{3} GSFP_{b_q}^{p_g} \cdot P_q^g - PLimit_b^p \right) \\
& - \sum_{b=1}^{B} \sum_{p=1}^{3} \mu_b^{p-} \left(- \sum_{q=1}^{N} \sum_{g=1}^{3} GSFP_{b_q}^{p_g} \cdot P_q^g - PLimit_b^p \right) \\
& - \sum_{p=1}^{2} \sum_{m=2, m\neq p}^{3} \mu^{pm+} \left(\sum_{n=2}^{N} P_n^p - \sum_{n=2}^{N} P_n^m - \gamma \right) \\
& - \sum_{p=1}^{2} \sum_{m=2, m\neq p}^{3} \mu^{pm-} \left(- \sum_{n=2}^{N} P_n^p + \sum_{n=2}^{N} P_n^m - \gamma \right)
\end{aligned} \tag{2.55}$$

LMP at node i with phase g can be derived by differentiating Lagrangian function (2.55) with respect to fixed load at node i phase g. λ^p is the Lagrange multiplier of real power balance constraint of phase p (40); μ_b^{p+} and μ_b^{p-} are the Lagrange multipliers of distribution line thermal limit constraints (44); μ^{pm+} and μ^{pm-} are the Lagrange multipliers of phase imbalance constraints (4).

As shown in (46), three-phase LMPs can be decomposed into four component: marginal energy component, marginal loss component, marginal congestion component, and marginal phase imbalance component. Compared with single-phase LMP, three-phase LMP has an extra component, namely, marginal phase imbalance component.

$$\begin{aligned} LMP_i^g &= \frac{\partial \psi}{\partial FD_i^g} \\ &= \sum_{p=1}^{3} \lambda^p (DF_i^g)^p + \sum_{b=1}^{B} \sum_{p=1}^{3} \mu_b^{p\prime} GSFP_{b\_i}^{p\_g} + \mu^{g\prime\prime} \\ &= \lambda^g - \sum_{p=1}^{3} \lambda^p (LF_i^g)^p + \sum_{b=1}^{B} \sum_{p=1}^{3} \mu_b^{p\prime} GSFP_{b\_i}^{p\_g} + \mu^{g\prime\prime} \end{aligned} \tag{2.56}$$

where

$$\mu_b^{p\prime} = \mu_b^{p+} - \mu_b^{p-} \tag{2.57}$$

$$\mu^{g\prime\prime} = \begin{cases} \mu^{12+} + \mu^{13} - \mu_i^{12-} - \mu^{13-} & \text{if } g = 1; \\ -\mu^{12+} + \mu^{23} + \mu_i^{12-} - \mu^{23-} & \text{if } g = 2; \\ -\mu^{13+} - \mu^{23} + \mu_i^{13-} + \mu^{23-} & \text{if } g = 3. \end{cases} \tag{2.58}$$

2.2.2 Simulation Results

A. A comparison between three-phase DCOPF and three-phase ACOPF

The simulation results of three-phase DCOPF and three-phase ACOPF are shown in Table 2.1 when the voltage of the reference bus is at 1.0 per unit. As shown in Table 2.1, the differences in social welfare, system real power losses, and real power line flows are very small between the proposed three-phase DCOPF and the benchmark three-phase ACOPF algorithm. The differences in reactive power flow are slightly larger than the real power

flow due to the fact that reactive power losses are not modeled in the three-phase DCOPF algorithm. However, as the power factors of distribution loads are typically around 0.95 lagging, the errors of reactive power flow are usually not very significant.

Table 2.1: Comparison between three-phase DCOPF and three-phase ACOPF

	DCOPF	ACOPF
Social Welfare ($)	833.4	838.6
Real Power Loss (KW)	38.9	47.1
Power Flows: Line 1 (KVA)	613+242.2i	618.6+281.6i
	631.4+242.2i	633.6+281.5i
	654.6+242.2i	654.9+288.7i
Power Flows: Line 2 (KVA)	611.5+242.2i	617.2 + 279.5i
	629.9+242.i	632.7+279.5i
	652.8+242.2i	653.9+286.2i
Power Flows: Line 3 (KVA)	605.2+242.i	614.9+265.7i
	624.4+242.2i	630.3+265.1i
	645.7+242.2i	651.3+270.8i

Numerical errors of real power losses and total social welfare are calculated for the proposed three-phase DCOPF algorithm. Figure 2.1. depicts the change in the numerical errors with various load bus voltage levels. The proposed three-phase DCOPF algorithm achieves best accuracy when the load bus three-phase average voltage is around 1.05 per unit. If the load bus voltage is kept between 0.98 and 1.02 per unit, then the errors of social welfare and real power losses associate with the three-phase DCOPF algorithm are smaller than 0.5% and 10% respectively.

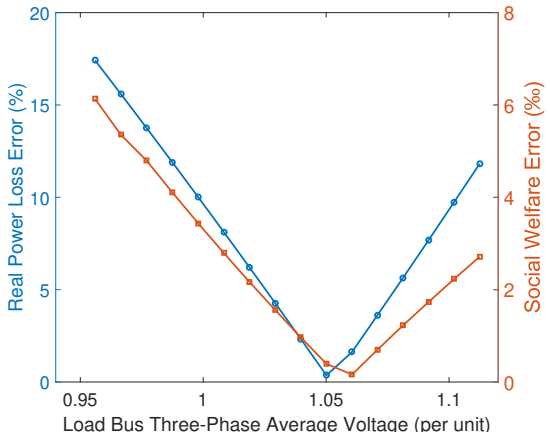

Figure 2.1: Numerical error versus load bus voltage

B. Three-phase LMP decomposition

The LMPs of the IEEE 4-bus network are shown in Table III. As illustrated in equation (46), three-phase LMPs consists of marginal energy component, marginal loss component, marginal congestion component, and marginal phase imbalance component. However, because the congestion constraints and phase imbalance constraints are not binding, these two components are not present in Table 2.2. The marginal energy components are \$0.6/kWh for every single bus and phase in the network. The marginal loss components and loss factors increase from distribution substation to the end of the feeder. The marginal loss components are higher on phase c whose loads are slightly higher than that of phase a and b. In order to show the effects of marginal phase imbalance component, simulations are performed by setting the fixed load of phase a as 460 KW and the fixed load of phase c as 530KW. The phase imbalance limit is set as 50 KW. The result of LMPs is shown in Table 2.3. The phase imbalance constraint relating phase a and c is binding. The marginal

imbalance price component of phase a is about \$-0.1/kWh, while the marginal imbalance price component of phase c is about \$0.1/kWh. Phase imbalance components of three-phase LMPs are crucial economic signals sent to customers on phase a and c instructing them to adjust load level to alleviate phase imbalance problems. The effect of congestion component is intuitive and straightforward.

Table 2.2: Three-phase LMPs with only energy and loss components

Price (\$/KWh)	Node 2	Node 3	Node 4
Phase A	$0.6 + 0.0016$	$0.6 + 0.0053$	$0.6 + 0.0234$
Phase B	$0.6 + 0.0013$	$0.6 + 0.0051$	$0.6 + 0.02$
Phase C	$0.6 + 0.0016$	$0.6 + 0.0055$	$0.6 + 0.0228$

Table 2.3: Three-phase LMPs with energy, loss, and phase imbalance components

		Price (\$/KWh)
Node2	Phase A	$0.6 + 0.0016 - 0.1033$
	Phase B	$0.6 + 0.0052 - 0.1033$
	Phase C	$0.6 + 0.0228 - 0.1033$
Node3	Phase A	$0.6 + 0.0013 + 0$
	Phase B	$0.6 + 0.0051 + 0$
	Phase C	$0.6 + 0.0200 + 0$
Node4	Phase A	$0.6 + 0.0015 + 0.1033$
	Phase B	$0.6 + 0.0054 + 0.1033$
	Phase C	$0.6 + 0.0226 + 0.1033$

2.2.3 Conclusion

This work developed a three-phase iterative DCOPF algorithm with fictitious nodal demand. GSF matrix, LF and DF are derived within the three-phase DCOPF frame-

work. The derivation for three-phase LMP decomposition shows that LMP can be decomposed into four price components: marginal energy component, marginal loss component, marginal congestion component and marginal phase imbalance component. Simulation results from the IEEE 4-bus test case demonstrated the validity of the proposed three-phase DCOPF algorithm. The three-phase DCOPF algorithm is shown to be a good approximation of the ACOPF algorithm when the load bus voltage is within normal operating range.

2.3 Three-phase ACOPF

2.3.1 Problem Formulation

The SDP formulation of single-phase alternating current OPF (ACOPF) problem was derived with voltages in rectangular form [48]. Reference [55] extended the SDP formulation to three-phase ACOPF problem with complex voltages. In this section, we formulate the three-phase ACOPF problem with voltages in the rectangular form.

List of Symbols

E_k^p, F_k^p Real part and imaginary part of the voltage at node k with phase p.

e_i Standard basis vector.

g_{ik}^{pm}, b_{ik}^{pm} Conductance and susceptance between node i with phase p and node k with phase m in the line admittance matrix.

G_{ik}^{pm}, B_{ik}^{pm} Conductance and susceptance between node i with phase p and node k with

phase m in the admittance matrix.

G The set of nodes with controllable generations in the power distribution network.

M_k^p Matrix defined for voltage magnitude calculation for node i with phase p.

N The set of all nodes in the power network.

N_A Total number of decomposed areas.

n_s Number of areas to search for further partitions at the current stage.

$P_{D_k}^p, Q_{D_k}^p$ Fixed real and reactive load at node k with phase p.

$P_{G_k}^p, Q_{G_k}^p$ Fixed real and reactive power generation at node k with phase p.

$P_{k,inj}^p, Q_{k,inj}^p$ Real and reactive power injection at node k with phase p.

$\underline{P}_k^p, \overline{P}_k^p$ Lower and upper limit of real power capacity of controllable distributed generation at node k with phase p.

P_{ik}^{pm}, Q_{ik}^{pm} Real and reactive power flow from node i with phase p to node k with phase m.

$\underline{Q}_k^p, \overline{Q}_k^p$ Lower and upper limit of reactive power capacity of controllable distributed generation at node k with phase p.

V Nodal voltage vector.

V_k^p Voltage at node k with phase p .

$\underline{V}_k^p, \overline{V}_k^p$ Lower and upper limit of voltage magnitude at node k with phase p.

X_l^{ext} Sub-matrix of X associated with nodes in the l-th extended sub-area.

$X_l^{ext(r)}$ Sub-matrix of X_l^{ext} associated with nodes in the l-th extended sub-area intersected with the r-th extended sub-area.

y_{ik}^{pm} Line admittance between node i with phase p and node k with phase m.

Y Admittance matrix.

$\mathbf{Y}_k^p, \overline{\mathbf{Y}}_k^p$ Admittance matrices defined for real and reactive power injection calculation.

$\mathbf{Y}_{ik}^p, \overline{\mathbf{Y}}_{ik}^p$ Admittance matrices defined for real and reactive power flow calculation.

$\mathbf{Y}_k^{p(l)}, \overline{\mathbf{Y}}_k^{p(l)}$ Admittance matrices defined for real and reactive power injection calculation associated with the branches in the l-th extended area.

$\mathbf{Y}_{ik}^{p\ (l)}, \overline{\mathbf{Y}}_{ik}^{p\ (l)}$ Admittance matrices defined for real and reactive power flow calculation associated with the nodes in the l-th extended area.

A. Matrix Definition

For a three-phase n-node distribution network, define the voltage vector as:

$$V \triangleq [E_1^1, E_1^2, E_1^3 \cdots E_n^1, E_n^2, E_n^3, F_1^1, F_1^2, F_1^3 \cdots F_n^1, F_n^2, F_n^3]^T$$

where E_k^p and F_k^p are the real and imaginary parts of complex voltage at node k with phase p.

Define the matrix Ψ_k^p as

$$\Psi_k^p \triangleq e_{3(k-1)+p} e_{3(k-1)+p}^T Y \tag{2.59}$$

where Y is the admittance matrix of the distribution network [55] and $e_{3(k-1)+p}$ is the standard basis vector with the $[3(k-1)+p]$-th element being 1, the only non-zero entry.

$$e_{3(k-1)+p} \triangleq [0, 0, 0, ..., 1, 0, ..., 0]^T \tag{2.60}$$

Define the admittance matrices to be used for power injection calculations as:

$$\mathbf{Y}_k^p \triangleq \frac{1}{2} \begin{bmatrix} Re(\Psi_k^p + \Psi_k^{pT}) & Im(\Psi_k^{pT} - \Psi_k^p) \\ Im(\Psi_k^p - \Psi_k^{pT}) & Re(\Psi_k^p + \Psi_k^{pT}) \end{bmatrix} \tag{2.61}$$

$$\overline{\mathbf{Y}}_k^p \triangleq -\frac{1}{2} \begin{bmatrix} Im(\Psi_k^p + y_k^{p\prime T}) & Re(\Psi_k^p - \Psi_k^{pT}) \\ Re(\Psi_k^{pT} - y_k^{p\prime}) & Im(\Psi_k^p + \Psi_k^{pT}) \end{bmatrix} \tag{2.62}$$

Then the real and reactive power injection equations can be rewritten as follows:

$$P_{k,inj}^p = Tr\{\mathbf{Y}_k^p V V^T\} \tag{2.63}$$

$$Q_{k,inj}^p = Tr\{\overline{\mathbf{Y}}_k^p V V^T\} \tag{2.64}$$

Define the admittance matrices Ψ_{ik}^p, $\mathbf{Y}_{ik}^p$, and $\overline{\mathbf{Y}}_{ik}^p$ to be used for branch flow calculations as follows:

$$\Psi_{ik}^p \triangleq e_{3(i-1)+p} \sum_{m=1}^{3} (e_{3(i-1)+m} \cdot y_{ik}^{pm} - e_{3(k-1)+m} \cdot y_{ik}^{pm})^T$$

$$\mathbf{Y}_{ik}^p \triangleq \frac{1}{2} \begin{bmatrix} Re(\Psi_{ik}^p + \Psi_{ik}^{p\ T}) & Im(\Psi_{ik}^{p\ T} - \Psi_{ik}^p) \\ Im(\Psi_{ik}^p - \Psi_{ik}^{p\ T}) & Re(\Psi_{ik}^p + \Psi_{ik}^{p\ T}) \end{bmatrix} \tag{2.65}$$

$$\overline{\mathbf{Y}}_{ik}^{p} \triangleq -\frac{1}{2}\begin{bmatrix} Im(\Psi_{ik}^{p}+\Psi_{ik}^{p\ T}) & Re(\Psi_{ik}^{p}-\Psi_{ik}^{p\ T}) \\ Re(\Psi_{ik}^{p\ T}-\Psi_{ik}^{p}) & Im(\Psi_{ik}^{p}+\Psi_{ik}^{p\ T}) \end{bmatrix} \tag{2.66}$$

where y_{ik}^{pm} is the line admittance between node i with phase p and node k with phase m.

Then the branch power flow connecting node i and node k with phase p can be rewritten as follows:

$$S_{ik}^{p} = Tr\{\mathbf{Y}_{ik}^{p} VV^{T}\} + jTr\{\overline{\mathbf{Y}}_{ik}^{p} VV^{T}\} \tag{2.67}$$

Define matrix M_i^p as:

$$M_{k}^{p} \triangleq \begin{bmatrix} e_{3(k-1)+p}e_{3(k-1)+p}^{T} & 0 \\ 0 & e_{3(k-1)+p}e_{3(k-1)+p}^{T} \end{bmatrix} \tag{2.68}$$

Then the square of voltage magnitude can be rewritten as:

$$|V_{i}^{p}|^{2} = Tr\{M_{k}^{p} VV^{T}\} \tag{2.69}$$

B. Three-phase ACOPF Problem

The objective of the three-phase ACOPF problem in a distribution system is to maximize total social welfare, minimize total power purchase cost, or minimize distribution system losses. The three-phase OPF problem can be formulated in $X = VV^T$ with matrices defined in section II.A as follows:

Formulation 1:

$$\min_{X} C(X) \tag{2.70}$$

subject to:

$$P_{G_k}^{p} - P_{D_k}^{p} = Tr\{\mathbf{Y}_{k}^{p} X\}, \quad k \in N \setminus G \tag{2.71}$$

$$Q^p_{G_k} - Q^p_{D_k} = Tr\{\overline{\mathbf{Y}}^p_k X\}, \quad k \in N \setminus G \tag{2.72}$$

$$\underline{P}^p_k - P^p_{D_k} \leq Tr\{\mathbf{Y}^p_k X\} \leq \overline{P}^p_k - P^p_{D_k}, \quad k \in G \tag{2.73}$$

$$\underline{Q}^p_k - Q^p_{D_k} \leq Tr\{\overline{\mathbf{Y}}^p_k X\} \leq \overline{Q}^p_k - Q^p_{D_k}, \quad k \in G \tag{2.74}$$

$$Tr\{\mathbf{Y}^p_{ik} X\}^2 + Tr\{\overline{\mathbf{Y}}^p_{ik} X\}^2 \leq (S^{p\ max}_{ik})^2, \quad i,k \in N \tag{2.75}$$

$$(\underline{V}^p_k)^2 \leq Tr\{M^p_k X\} \leq (\overline{V}^p_k)^2, \quad k \in N \tag{2.76}$$

$$X = VV^T \tag{2.77}$$

Equation (2.77) is equivalent to the following two equations:

$$X \succeq 0 \tag{2.78}$$

$$rank(X) = 1 \tag{2.79}$$

In this work, the objective function (2.70) is chosen to minimize the total power purchase cost.

$$C(X) = \sum_{p=1}^{3} \sum_{k \in G} c^p_k P^p_k \tag{2.80}$$

where c^p_k and P^p_k are the supply offer price and generation quantity at controllable generation node k with phase p. Equations (2.71) and (2.72) enforce real and reactive power balance constraints for load buses. Equations (2.73) and (2.74) represent real and reactive power generation capacity constraints for buses with distributed generations. Power flow constraints are modeled in equation (2.75). Voltage constraint is enforced in equation (2.76). Equation (2.77) can be replaced by a positive semidefinite constraint and a rank constraint. The rank constraint is a non-convex constraint.

2.3.2 Technical Approach

A. Rank-one Conundrum Revisited

The rank constraint makes it difficult to solve the reformulated OPF problem. Many researchers tried to solve the OPF problem by applying the semidefinite relaxation technique in which the rank constraint is dropped [46, 48, 50, 58, 59, 55]. Some heuristic methods were developed to recover a rank-one solution for single-phase networks when the semidefinite relaxation technique fails[51, 52]. The existence of global optimal rank-one solution has been proved for single-phase radial network in [47]. It is claimed in [55] that semidefinite relaxation is "exact" for the three-phase OPF problem in a radial network. However, no rigorous proof was provided. The semidefinite relaxation technique did result in global optimal solution in the numerical tests [55]. However, the results are obtained when the supply offer prices of the three phases are exactly the same. In practice, it is not realistic to assume that the supply offer prices from distributed energy resources on three phases will be the same [8]. A counter example is given in this subsection to prove that semidefinite relaxation is not "exact" for three-phase OPF problems.

In order to prove the "exactness" of SDP relaxation for OPF problems of single-phase tree-networks, the geometry of the feasible power injection region is analyzed [47]. Similarly, the feasible power injection region of a three-phase two-node network is studied here. It can be assumed that a network consists of two three-phase nodes connected by a typical distribution line. Define P_1^1, P_1^2, P_1^3 and P_2^1, P_2^2, P_2^3 as the power injections of the three phases at node 1 and node 2 respectively. Assuming the voltage magnitudes are around 1 per unit, the power injections can be calculated with the differences in voltage

angles.

The 2-bus three-phase network is analyzed under two scenarios. In the first scenario, it is assumed that the supply offer prices of DERs are the same for all three phases. Then, the OPF problem is equivalent to optimize over the feasible injection region of power summed over three phases. The feasible region on the plane of power injection at node 1 versus node 2, i.e., P_1 versus P_2, is depicted in Figure 2.2.

In the second scenario, it is assumed that the supply offer prices are different on the three phases. In this case, the above-mentioned equivalence is no longer valid. The projection of the six-dimensional feasible power injection region onto P_1^1 versus P_2^1 plane for the second scenario is depicted in Figure 2.3.

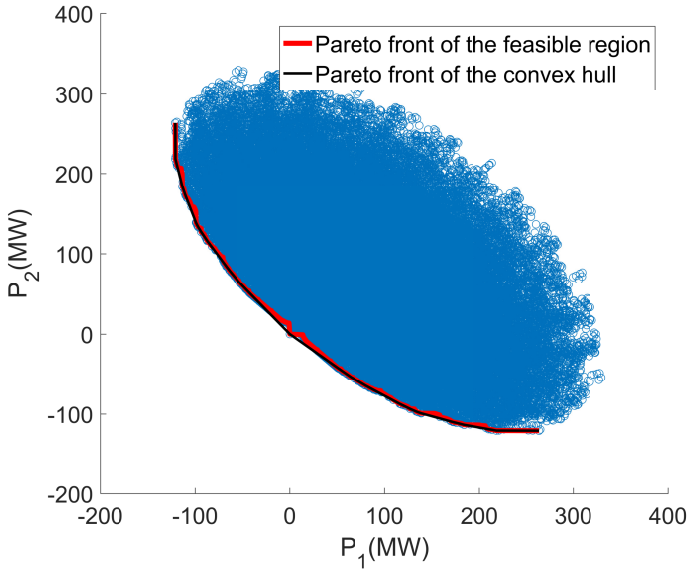

Figure 2.2: Feasible power injection region of a two-node network with the same supply offer prices on three phases

In the first scenario, the supply offer prices of the three different phases are the

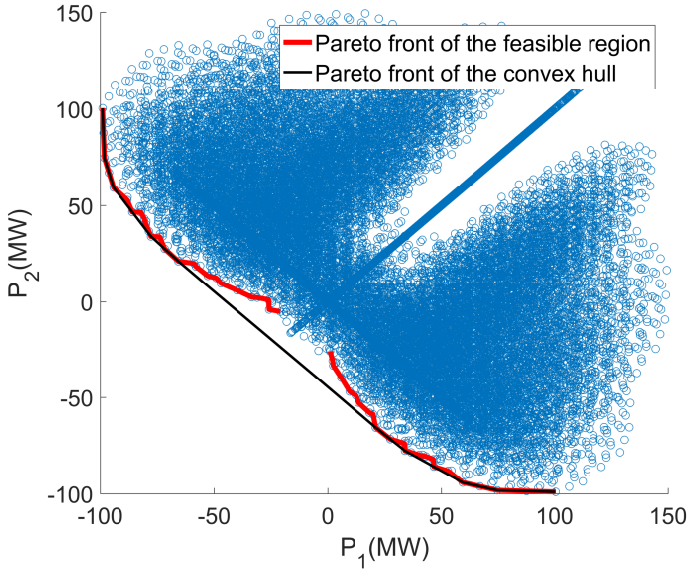

Figure 2.3: Feasible power injection region of a two-node network with different supply offer prices on three phases

same. Therefore, the feasible power injection region on the P_1 and P_2 plane is approximately an ellipsoid as shown in Figure 2.2. By dropping the rank constraint, the new feasible region can be obtained by taking the convex hull of the original region, which is the same ellipsoid. The optimal solution is located on the Pareto front of the feasible power injection region. Therefore, relaxing the rank constraint doesn't influence the optimal solution as the Pareto front of the two feasible power injection regions are the same.

In the second scenario, the supply offer prices of DERs on three phases are different. The projection of the feasible power injection region onto the P_1^1 and P_2^1 plane is non-convex as shown in Figure 2.3. Taking convex hull will enlarge the original feasible region. Therefore, the Pareto front of the relaxed problem is different and the solutions with semidefinite relaxation technique will have higher ranks. To resolve the rank conundrum, we advocate the adoption of the convex iteration technique to solve the three-phase OPF

problem.

B. Convex Iteration

Instead of directly dropping the rank constraint in Formulation 1, we advocate the adoption of the convex iteration technique to express the rank-constrained optimization problem as iteration of the convex problem sequence (2.81) and (2.82) [60] in Formulation 2:

Formulation 2:

$$\min_{X} C(X) + w\,Tr(XW^*)$$

subject to

$$\begin{gathered} X \in B \\ X \succeq 0 \\ \min_{W \in S^{N_X}} Tr(X^*W) \end{gathered} \tag{2.81}$$

subject to

$$\begin{gathered} 0 \preceq W \preceq I \\ Tr(W) = N_X - 1 \end{gathered} \tag{2.82}$$

where B denotes the feasible region of X defined by equations $(13) - (18)$, W^* represents the optimal solution to semidefinite program (2.82), and X^* denotes the optimal solution to semidefinite program (2.81). The size of X and W are both $N_X \times N_X$. The closed-form solution of the second convex optimization problem (2.82) is

$$W = U(:, 2 : N_X)U(:, 2 : N_X)^T \tag{2.83}$$

where U can be obtained from the eigenvalue decomposition

$$X^* = U\Lambda U^T \tag{2.84}$$

The result achieved from SDP relaxation can be used as a starting point, as the convex hull usually provides a tight lower bound. The initial value of the direction matrix can be chosen as the zero matrix. The optimal direction matrix W^*_{opt} is defined as any positive semidefinite matrix yielding optimal solution X^* of rank one. Therefore , the following two problems are equivalent when W^*_{opt} is found.

$$\begin{aligned} \min \quad & C(X) + w\,Tr(XW^*_{opt}) \\ s.t. \quad & X \in B \\ & X \succeq 0 \end{aligned} \quad \equiv \quad \begin{aligned} \min \quad & C(X) \\ s.t. \quad & X \in B \\ & X \succeq 0 \\ & rank(X) = 1 \end{aligned}$$

It should be noted that the convex iteration algorithm is different from the relaxation of the rank-constrained optimization problem. However, at global optimality, the convex iteration formulation is equivalent to the relaxed problem. The convex iteration algorithm was successfully applied in other applications including sensor-network localization and compressed sensing [60]. By contrast, penalization methods [53, 52] tries to recover a rank-one solution from the lower bound of the optimal solution by minimizing either the voltage difference or reactive power loss.

C. Intuition of Convex Iteration Algorithm

The derivation of the convex iteration algorithm can be intuitively explained as follows. For a rank-one positive semidefinite matrix, the largest eigenvalue is also the only

non-zero eigenvalue, i.e.,

$$Tr(X) = \sum_i \lambda(X)_i = \lambda_{max}(X) \tag{2.85}$$

The rank-one constraint is equivalent to the following constraint:

$$Tr(X) - \lambda_{max}(X) = 0 \tag{2.86}$$

where the largest eigenvalue can be obtain by:

$$\max_{||u||_2=1} u^T X u$$

Therefore, constraint (2.86) can be rewritten as:

$$\{\min_{||u||_2=1} Tr(X(I - uu^T))\} = 0$$

This is equivalent to:

$$\{\min_{I \succeq W \succeq 0} Tr(XW)\} = 0 \tag{2.87}$$

By multiplying the equality constraint (2.87) with w and adding it to the objective function (2.85), the original problem in Formulation 1 can be rewritten as:

$$\min_{X,W} C(X) + w\, Tr(XW)$$

subject to

$$\begin{gathered} X \in B \\ X \succeq 0 \\ I \succeq W \succeq 0 \end{gathered} \tag{2.88}$$

As shown in the above optimization problem formulation, the rank-one constraint is re-expressed as a bilinear term in the objective function. For a traditional bilinear optimization problem, iterative linear programming method can be applied to find the optimal

solution(s). In the context of semidefinite programming, the optimization problem (2.88) can be tackled by iteratively solving the convex problem sequence (2.81) and (2.82).

The meaning of the direction matrix W can also provide us some intuition about the inner working of the convex iteration algorithm. Let's define matrix subspace $\mathcal{S}_n$ as

$$\mathcal{S}_n \triangleq \{(I-W)X(I-W)|X \in S_+^N\} \tag{2.89}$$

It can be shown that the orthogonal compliment of $\mathcal{S}_n$ is

$$\mathcal{S}_n^{\perp} = \{WXW|X \in S_+^N\} \tag{2.90}$$

The optimal solution to semidefinite program (2.81) X^* can be decomposed into two components (2.91). The first component is the projection of X^* onto subspace $\mathcal{S}_n$, which is $(I-W)X^*(I-W)$. The second component is the projection of X^* onto subspace $\mathcal{S}_n^{\perp}$, which is WX^*W.

$$X^* = (I-W)X^*(I-W) + WX^*W \tag{2.91}$$

According to Eckart-Young Thereom, the best rank-one approximation of X^* in terms of Frobenius norm distance is:

$$\hat{X^*} = U(:,1)\Lambda(1,1)U(:,1)^T \tag{2.92}$$

It can be shown that the following equality holds:

$$(I-W)X^*(I-W) = U(:,1)\Lambda(1,1)U(:,1)^T \tag{2.93}$$

Therefore, the projection of X^* onto $\mathcal{S}_n$ is $\hat{X^*}$. Hence, polar direction $-W$ can be regarded as pointing toward the set of all rank-1 positive semidefinite matrices whose nullspace contains that of X^*.

D. Chordal Conversion

Coming back to Formulation 1 and first setting aside the rank constraint, a SDP programming problem needs to be solved. In most of the SDP solvers, the primal-dual interior-point method is adopted. The disadvantage of the primal-dual interior-point method is that it is time-consuming to construct the dense Schur complement matrix when solving large-scale problems. To address this drawback, the underlying aggregated sparsity of the power network is exploited by researchers [58, 59]. The semidefinite completion theory allows us to exploit the chordal sparsity of radial distribution networks [61]. The semidefinite completion theorem states that a symmetric matrix is positive semidefinite completable if and only if all of the small matrices associated with the maximal cliques of the graph derived from the whole matrix are positive semidefinite. This property allows the SDP problem to be converted into another form with smaller-sized positive semidefinite variables. The details of the conversion method are described in [62, 63]. When decomposing the graph of large networks, the intersections of maximal cliques are not empty. Thus, equality constraints of the intersection areas are introduced which may increase the dimension of the Schur complement matrix. Decisions need to be made to determine the trade-off between the sparsity and order of the Schur complement matrix. Some heuristic algorithms of clique amalgamation were developed in [62, 59].

By adopting chordal conversion, the original three-phase OPF problem (Formulation 1) can be reformulated as follows:

Formulation 3:

$$\min_{X} \sum_{l=1}^{N_A} C_l \left(X_l^{ext} \right) \tag{2.94}$$

subject to:

$$X_l^{ext} \in B^{(l)}, \quad l = 1, 2, \cdots N_A \tag{2.95}$$

$$X_l^{ext(r)} = X_r^{ext(l)}, \quad l, r = 1, 2, \cdots N_A \tag{2.96}$$

$$X_l^{ext} \succeq 0, \quad l = 1, 2, \cdots N_A \tag{2.97}$$

$$rank(X_l^{ext}) = 1, \quad l = 1, 2, \cdots N_A \tag{2.98}$$

where $B^{(l)}$ is the feasible region of X_l^{ext} satisfying

$$P_{G_k}^p - P_{D_k}^p = Tr\{\mathbf{Y}_k^{p(l)} X_l^{ext}\}, \quad k \in A_l$$

$$Q_{G_k}^p - Q_{D_k}^p = Tr\{\overline{\mathbf{Y}}_k^{p(l)} X_l^{ext}\}, \quad k \in A_l$$

$$\underline{P}_k^p - P_{D_k}^p \leq Tr\{\mathbf{Y}_k^{p(l)} X_l^{ext}\} \leq \overline{P}_k^p - P_{D_k}^p, k \in A_l \setminus G$$

$$\underline{Q}_k^p - Q_{D_k}^p \leq Tr\{\overline{\mathbf{Y}}_k^{p(l)} X_l^{ext}\} \leq \overline{Q}_k^p - Q_{D_k}^p, k \in A_l \setminus G$$

$$Tr\{\mathbf{Y}_{ik}^{p\ (l)} X_l^{ext}\}^2 + Tr\{\overline{\mathbf{Y}}_{ik}^{p\ (l)} X_l^{ext}\}^2 \leq (S_{ik}^{p\ max})^2, k \in A_l \cap G$$

$$(\underline{V}_k^p)^2 \leq Tr\{M_k^{p(l)} X\} \leq (\overline{V}_k^p)^2, \quad k \in A_l$$

A_l is the set of nodes in the l-th area. A_l^{ext} denotes the set of nodes in the l-th extended area which is defined as the union of A_l and the nodes of the other areas directly connected to the l-th area. A more detailed description of the extended area concept is provided in [55]. V_l^{ext} denotes the voltage vector with the nodes in A_l^{ext}. $X_l^{ext(r)}$ is the sub-matrix of X_l^{ext} collecting the columns and rows of X_l^{ext} corresponding to the voltages of $A_l^{ext(r)} = \{A_l^{ext} \cap A_r^{ext}\}$.

The decomposition of rank constraint (2.79) is obvious. If matrix X is rank-one, then all the sub-matrices X_l^{ext} are rank-one. If all the sub-matrices X_l^{ext} are rank-one, then voltage vector V can be constructed with the results of singular value decomposition of all sub-matrices X_l^{ext}. Consequently, matrix X can be obtained from V.

E. Chordal Conversion Based Convex Iteration

By synergistically combining the chordal conversion method and the convex iteration technique, we propose a new iterative three-phase OPF solution algorithm as follows.

Formulation 4: Step 1:

$$\min_{X} \sum_{l=1}^{N_A} C_l \left(X_l^{ext}\right) + \sum_{l=1}^{N_A} w_l \, Tr(X_l^{ext} W_l^*) \tag{2.99}$$

s.t.

$$X_l^{ext} \in B^{(l)}, \quad l = 1, 2, \cdots N_A \tag{2.100}$$

$$X_l^{ext(r)} = X_r^{ext(l)}, \quad l, r = 1, 2, \cdots N_A \tag{2.101}$$

$$X_l^{ext} \succeq 0, \quad l = 1, 2, \cdots N_A \tag{2.102}$$

Step 2:

$$W_l = U_j(:, 2 : N_{X_l^{ext}}) \, U_j(:, 2 : N_{X_l^{ext}})^T \tag{2.103}$$

where the size of X_l^{ext} is $N_{X_l^{ext}} \times N_{X_l^{ext}}$. U_j is obtained from the singular value decomposition.

$$X_j^{ext} = U_j \Lambda_j U_j^T \tag{2.104}$$

At a global optimum where the trace regularization term equals to zero, Formulation 4 becomes the convex equivalent of Formulation 3. The feasible set in Formulation 4

contains all rank-one symmetric matrices. An optimal rank-one solution X^*_{opt} from Formulation 4 will also minimize the objective function of Formulation 3.

The convergence to global optimality from an arbitrary initial point is not guaranteed. However, the algorithm will always arrive at a stalling point when the trace regularization term no longer decreases due to the monotonically non-increasing objective function sequence [60]. To re-start the algorithm with new search directions, the randomization technique can be leveraged [60]. Specifically with rank-one constraints, the direction matrices can be reinitialized as:

$$W_l = U_j(:, 2 : N_{X_l^{ext}}) \, (U_j(:, 2 : N_{X_l^{ext}})^T + rand(N_{X_l^{ext}} - 1, 1) U_j(:, 1)^T) \tag{2.105}$$

However, the re-start process may fail by converging to another or the same stalling point with a rank larger than one. In addition, the re-start process could make the algorithm much more time-consuming.

F. Greedy Partition of the Grid

The computational efficiency of the chordal conversion based convex iteration algorithm depends heavily on the choice of grid partition scheme. This subsection develops a greedy algorithm to find an appropriate grid partition scheme. The algorithm development is motivated by the relationship between the computational complexity of the SDP problem in the first step of Formulation 4 and the nonzero elements in the search direction matrix.

A closer look is taken firstly at the computational efficiency of the interior point method which is adopted by most of the existing SDP solvers including SeDuMi [64], SDPA, and MOSEK. As SeDuMi is one of the most popular open source SDP solver package, it is

chosen for illustration purpose in this subsection.

The SDP in Formulation 4 is first transformed to the standard conic form. The standard conic form of the SDP in formulation 4 can be written as follows.

$$\min c^T x \tag{2.106}$$

$$Ax = b \tag{2.107}$$

$$x \in \mathcal{S}^+ \tag{2.108}$$

where x is the vectorized primal variable and $\mathcal{S}^+$ is the semidefinite cone. In the primal-dual interior point method, scaling technique [65] is widely used. AHO [66], NT [67], and HKM [68] scaling are the most popular ones. With NT scaling adopted in SeDuMi, the scaling factor D [65, 69] is introduced to obtain the search direction in its iterative wide region method. Preconditioned gradient method is adopted in SeDuMi to obtain the inverse of ADA^T. In the preconditioning step, Cholesky decomposition of matrix ADA^T is performed. This is the most computationally expensive process in solving large-scale SDP problems. The computation cost of Cholesky decomposition heavily depends on the number of none-zero elements in matrix ADA^T. Therefore, to reduce the computation time, a greedy grid partition algorithm should search for the grid partition scheme which results in the least number of non-zero elements in matrix ADA^T.

For linear programming, the process of selecting the matrix ADA^T with the smallest number of non-zero elements can be accomplished by selecting matrix AA^T. This is because the sparsity patterns of the matrices ADA^T and AA^T are the same. Although this relationship doesn't hold for semidefinite programming, it still provides a good approxima-

Algorithm 2 Greedy algorithm for grid partition

Initialize $n_s = 1$

while 1 **do**

  if $n_s = 0$ **then**

    break

  else

    $n_{temp} = n_s$

    for $i = 1 : n_s$ **do**

      search for a possible cut in subarea i;

      if there exists a cut which reduces the size of AA^T **then**

        search along the edges in subarea i; find the cut which reduces the size of AA^T the most,

        $n_{temp} = n_{temp} + 1$ and record the edge as a cut.

      else

        subarea i is finalized, i.e.

        no more search will be performed in subarea i;

        $n_{temp} = n_{temp} - 1$.

      end if

    end for

    $n_s = n_{temp}$

  end if

end while

tion. In other words, as long as the sparsity pattern does not vary a lot among different grid decomposition schemes, the partition scheme with smaller-sized AA^T matrix is more computationally efficient in general. Based on this approximation, a greedy algorithm is developed to find the partition scheme which yields a AA^T matrix with the smallest size. The greedy algorithm can be carried out as in algorithm 1.

2.3.3 Numerical Study

The proposed chordal based convex iteration algorithm with greedy grid partition scheme is implemented in YALMIP [70]. Simulations are conducted on the IEEE 4-bus, 10-bus, 13-bus, 34-bus, 37-bus, 123-bus, and 906-bus three-phase test feeders to validate 1) the optimality and feasibility of the solutions from the proposed convex iteration algorithm, 2) the computational efficiency of the greedy grid partition scheme, and 3) the scalability of the chordal conversion based convex iteration algorithm. A Dell workstation with a 64-bit Intel Xeon Quad Core CPU at 3.30 GHz with 16 GB of RAM is used to perform the simulations.

A. Solution Optimality and Feasibility

The IEEE three-phase test feeders are modified to account for scenarios where the supply offer prices on the three phases are different. In the IEEE 4-bus test feeder, the loads on three phases at node 4 are set as $1800KW$, $1600KW$, and $1400KW$. The supply offer prices of the three phases are set as $\$1/KWh$, $\$0.5/KWh$, and $\$0.2/KWh$. Distribution generations are assumed to be located on node 4 with a generation capacity of $200KW$ per phase. In the 10-bus test feeder [55], the loads on the three phases are set as

$700KW$, $530KW$, and $600KW$ to create unbalanced scenario. The supply offer prices of the three phases are set as $\$1/KWh$, $\$0.3/KWh$, and $\$0.6/KWh$. Distributed generations are placed on node 5 and 7 with a generation capacity of $50KW$ per node per phase. In the IEEE 13-bus test feeder, the load profile on the three phases is kept the same as $1175KW$, $1039KW$, and $1252KW$. Distributed generations are placed on node 611, 652, 671, and 634 with a generation capacity of $50KW$ per node per phase. The supply offer prices of the three phases are set as $\$0.6/KWh$, $\$0.3/KWh$, and $\$1/KWh$. For the IEEE 34-bus test feeder, 50% load profile is adopted to avoid incorporating discrete control variables of the voltage regulators. The loads on the three phases are $303KW$, $292KW$, and $289.5KW$. The distributed generations are placed on node 814, 836, and 890 with generation capacity of $20KW$ per node per phase. The supply offer prices of three phases are set as $\$1/KWh$, $\$0.9/KWh$, and $\$0.8/KWh$. For the IEEE 37-bus test feeder, the distributed generations are placed on node 701, 704, 707, 711, 744, 730, and 734 with generation capacity of $50KW$ per node per phase. The supply offer prices of three phases are set as $\$0.6/KWh$, $\$0.3/KWh$, and $\$1/KWh$. For the IEEE 123-bus test feeder, the distributed generations are placed on node 7, 18, 25, 35, 44, 54, 72, 76, 89, 97, and 105, with generation capacity of $50KW$ per node per phase. The supply offer prices of three phases are set as $\$1/KWh$, $\$0.3/KWh$, and $\$0.6/KWh$. For the IEEE European LV test feeder with 906 nodes, the distributed generations are placed on node 145, 155, 391, 707, and 745 with generation capacity of $0.5KW$ per node per phase. The supply offer prices of three phases are set as $\$0.6/KWh$, $\$0.7/KWh$, and $\$0.5/KWh$.

To illustrate the optimality and feasibility of solutions under the proposed algo-

rithm, a comparison of the solutions obtained from traditional methods, including Powell method [71, 72] and interior-point method [73, 74], and the proposed convex iteration method is shown in Table 2.4. Additional test scenarios are created by varying the supply offer prices of the DERs.

Table 2.4: Comparison of traditional methods and the convex iteration method with different prices for DERs

Test system	Prices of three phases ($/kWh)	Objective value ($/hour)		
		Powell	Interior Point	Convex Iteration
4-bus	1/0.5/0.2	3121.9	3121.9	3121.9
test feeder	0.9/0.45/0.18	3091.9	3091.9	3086.9
10-bus	1/0.3/0.6	1229.2	1229.2	1229.1
test feeder	0.8/0.24/0.48	1191.4	1191.4	1191.3
13-bus	0.6/0.3/1	2345.4	2345.4	2345.4
test feeder	0.48/0.24/0.8	2290.2	2290.2	2290.2
34-bus	1/0.9/0.8	832.7	832.7	830.8
test feeder	0.9/0.81/0.72	816.5	816.5	815.4
37-bus	0.6/0.3/1	1740.3	1740.3	1739.5
test feeder	0.54/0.27/0.9	1675.9	1675.9	1675.4
123-bus	1/0.3/0.6	2414.6	2414.5	2413.6
test feeder	0.8/0.24/0.48	2205.6	2205.6	2205.0
906-bus	0.6/0.7/0.5	38.4	38.3	38.2
test feeder	0.54/0.63/0.45	37.9	37.9	37.7

As shown in Table 2.4, the proposed convex iteration approach achieves lower objective values on 11 out of 14 test scenarios. The traditional methods arrive at the same solution as the proposed convex iteration method on the other 3 test scenarios. As the size of the test feeder increases, it becomes more difficult for the traditional methods to match the performance of the proposed convex iteration algorithm.

To illustrate the optimality and feasibility of the proposed algorithm, another comparison of solutions derived from the SDP relaxation method [46, 48, 55] and the proposed convex iteration method with the default setting is shown in Table 2.5.

Table 2.5: Comparison of the SDP relaxation method and the convex iteration method with different prices for three phases

Test system	Method	Rank of solution	Objective value ($/hour)
4-bus	SDP relaxation	3	3085.6
test feeder	convex iteration	1	3121.9
10-bus	SDP relaxation	7	1216.3
test feeder	convex iteration	1	1229.1
13-bus	SDP relaxation	3	2319.5
test feeder	convex iteration	1	2345.4
34-bus	SDP relaxation	6*	831.8
test feeder	convex iteration	1	830.8
37-bus	SDP relaxation	1*	1739.5
test feeder	convex iteration	1	1739.5
123-bus	SDP relaxation	6*	2413.6
test feeder	convex iteration	1	2413.6
906-bus	SDP relaxation	6*	38.2
test feeder	convex iteration	1	38.2

It can be seen from Table 2.5 that the SDP relaxation method does not yield a rank-one solution by directly removing the rank constraint. For the IEEE 4-bus, 10-bus and 13-bus test feeders, the grids do not need to be partitioned. For the IEEE 34-bus, 37-bus, 123-bus, and 906-bus test feeders, the same grid partition scheme is adopted for both the SDP relaxation and the proposed convex iteration methods. The star symbol, $*$, represents the highest rank among all partitioned areas. The SDP relaxation method only succeeds in finding a feasible rank-one solution for the 37-bus test feeder. The high rank solutions

in other cases do not have any physical meaning. In most cases, the solution of the SDP relaxation method provides a lower bound of the original non-convex optimization problem. For the 34-bus test feeder, the SDP solver stops at a near-global optimal solution of the relaxed problem, which has a higher value than that of the convex iteration method. The numerical difficulty is caused by the extremely long and short distribution lines[75]. On the other hand, the proposed chordal conversion based convex iteration algorithm always produces a rank-1 solution.

Table 2.6: Comparison of the SDP relaxation method and the convex iteration method with same prices for three phases

Test system	Method	Rank of solution	Power loss (kW)
4-bus	SDP relaxation	1	325.9
test feeder	convex iteration	1	325.9
10-bus	SDP relaxation	1	12.2
test feeder	convex iteration	1	12.2
13-bus	SDP relaxation	1	89.4
test feeder	convex iteration	1	89.4
34-bus	SDP relaxation	6*	38.3
test feeder	convex iteration	1	37.5
37-bus	SDP relaxation	1	26.4
test feeder	convex iteration	1	26.4
123-bus	SDP relaxation	6*	32
test feeder	convex iteration	1	34.7
906-bus	SDP relaxation	7*	1.5
test feeder	convex iteration	1	1.3

If the prices are set to be $\$1/KWh$ for all three phases, the original problem is equivalent to minimization of the total power losses. As shown in Table 2.6, the SDP relaxation method is able to find the global optimum for the three small-scaled systems,

which is consistent with the analysis in section III.A and reference [55].

At last, a comprehensive comparison between the penalized SDP method [53] and the proposed convex iteration algorithm is conducted. The comparison results are shown in Table 2.7.

Table 2.7: Comparison of the penalized SDP method and the convex iteration method

Test system	Method	eig^2/eig^1	Power injection error (kW)
4-bus	penalized SDP	9.1×10^{-9}	5.6×10^{-3}
test feeder	convex iteration	2.6×10^{-9}	3.9×10^{-3}
10-bus	penalized SDP	7.7×10^{-7}	5.2×10^{-3}
test feeder	convex iteration	2.2×10^{-9}	6.8×10^{-3}
13-bus	penalized SDP	3.8×10^{-7}	0.2208
test feeder	convex iteration	3.2×10^{-9}	0.0629
34-bus	penalized SDP	1.2×10^{-5}	3.24
test feeder	convex iteration	6.0×10^{-8}	2.41
37-bus	penalized SDP	3.0×10^{-6}	1.54
test feeder	convex iteration	3.0×10^{-6}	1.54
123-bus	penalized SDP	2.8×10^{-5}	13.21
test feeder	convex iteration	1.2×10^{-8}	1.21
906-bus	penalized SDP	5.1×10^{-5}	6.7
test feeder	convex iteration	6.0×10^{-8}	2.3

For the IEEE 4-bus, 10-bus, and 13-bus test feeders, the comparison is performed without graph partition. Although the penalized SDP method did obtain a rank-one solution, the ratio of the second largest eigenvalue of matrix X to its largest eigenvalue is much larger than that of the proposed convex iteration method. Moreover, as shown in Table 2.7, the power injection error obtained from SVD of the rank-one solution of the penalized SDP method is much larger than that of the proposed convex iteration method.

For the IEEE 34-bus, 37-bus, 123-bus, and 906-bus test feeders, the comparison is performed with the same graph partition scheme obtained from the greedy algorithm. For IEEE 34-bus, 123-bus, and 906-bus test feeders, the penalized SDP method fails to find a rank-one solution. In IEEE 123-bus, one of the partitioned areas containing nodes 44, 47, 48, 49, and 50 is selected for verification. Under the penalized SDP method, the non-negligible eigenvalues of the variable matrix are 15.8738, 0.0004, 0.0004, 0.0002, 0.0002, and 0.0001. The p.u. complex voltage of the boundary node 50 obtained from rank-one approximation are different. The complex voltage of the boundary point under the penalized SDP method are $[1.0167 - 0.0283j, -0.5260 - 0.8964j, -0.5035 + 0.8964j]$ and $[1.0170 - 0.0280j, -0.5262 - 0.8969j, -0.5044 + 0.8967j]$ in two different extended areas. The power injection error under the penalized SDP method is also much larger than that of the proposed convex iteration method.

B. Effectiveness of the Greedy Grid Partition Scheme

Table 2.8: Computation time of IEEE 4-bus test feeder

Number of partition areas	Computation time (s)	Number of iterations	Number of Nonzero Elements
1	0.346	4	3.92×10^4
2	0.373	4	2.95×10^4
*2	0.373	4	2.95×10^4
3	0.484	4	3.63×10^4
4	0.577	4	4.25×10^4

To validate the effectiveness of the proposed greedy grid partition scheme, sim-

Table 2.9: Computation time of IEEE 13-bus test feeder

Number of partition areas	Computation time (s)	Number of iterations	Number of Nonzero Elements
1	68.397	20	2.12×10^6
2	10.789	14	5.39×10^5
3	9.659	15	4.22×10^5
*4	8.714	16	3.61×10^5
4	7.732	16	3.18×10^5
5	6.567	13	3.24×10^5
6	6.602	14	2.77×10^5
7	5.768	14	2.27×10^5
8	6.020	14	2.27×10^5
9	6.374	15	2.27×10^5
13	8.019	16	2.53×10^5

ulations are conducted on the IEEE 4-bus and 13-bus test feeders under all possible grid partition scenarios. An exhaustive search for all possible partition scenarios is conducted. The computation times of all scenarios are recorded. The results are then grouped by the number of partitioned areas. The computation times being reported in Table 2.8 and 2.9 are the shortest computation times for each number of partition areas using MOSEK. The computation times obtained from the greedy partition scheme is denoted by $*$.

It can be seen from Table 2.7 and 2.8 that the computation time of the proposed algorithm is approximately proportional to the number of nonzero elements in matrix ADA^T. The greedy algorithm successfully found grid partitioning schemes with very reasonable computation times. In the IEEE 4-bus test feeder, the computation time with the greedy partition scheme is almost the same as the shortest computation time found by exhaustive search. In the IEEE 13-bus test feeder, the computation time with the greedy partition

scheme is only 3 seconds longer than the shortest computation time.

C. Scalability of the Proposed Algorithm

As shown in Table 2.10, the computation times of the three-phase OPF problems on all seven IEEE test feeders is within 2 minutes using the entry level Dell workstation. The combination of the chordal based conversion technique and the greedy grid partition scheme made the proposed algorithm computationally efficient.

Table 2.10: Scalability of proposed algorithm

Test system	Computation time (s)	Number of iterations	Number of Nonzero Elements	Rank of Solution
4-bus	0.373	4	2.95×10^4	1
10-bus	12.127	29	2.53×10^5	1
13-bus	8.714	16	3.61×10^5	1
34-bus	4.161	3	1.25×10^6	1
37-bus	3.261	1	2.06×10^6	1
123-bus	27.182	3	4.93×10^6	1
906-bus	79.799	3	1.32×10^7	1

2.3.4 Conclusion

This work develops a chordal conversion based convex iteration algorithm to solve the three-phase OPF problem. A greedy grid partition scheme is also developed to improve the computational efficiency of the proposed algorithm. The simulation results show that the greedy algorithm can find an appropriate grid partition scheme which has similar computation time to that of the best partition found from the exhaustive search. At last, the scalability of the proposed algorithm is validated through simulations on the IEEE 123-bus and 906-bus test feeders. The proposed OPF algorithm can find the optimal solutions

within 2 minutes on an entry level Dell workstation. However, it should be noted that it is possible for the proposed convex iteration approach to converge to a local optimum and the re-start strategy may fail. Therefore, the proposed convex iteration algorithm does not guarantee convergence to global optimum solution(s) in all distribution feeders.

Chapter 3

Ancillary Service Provision with Data centers

As the digitization of our daily life, the generated data is growing explosively. The cloud computing gains more and more popularity for its flexibility and capacity, which has been used for various application including storage, database, and data analytics. To host the cloud computing services, new data centers are under continuously construction. Data centers have become one of the significant energy consumers, which consume about 2 % of the total energy consumption of the U.S. in 2014. The energy consumption of data centers can be adjusted through various techniques including dynamic voltage and frequency at chip level, sleep state at server level, virtual machine migration at data center level, and geo-distributed load balancing at grid level. In this chapter, the control flexibility of data centers is leveraged to provide ancillary services in electric grids, which can not only reduce the energy bills of data center but also improve grid stability and reliability.

3.1 Related Works

The existing research in the field of green computing tries to improve data center energy efficiency at five different levels. At the processor level, dynamic voltage and frequency scaling (DVFS) techniques have been shown to be highly effective in improving the energy-efficiency [76, 77, 78]. At the server level, various scheduling policies have been designed to create opportunities for deep sleep [79, 80]. At the data center level, virtual machine migration and autoscaling techniques have been proposed to optimize energy consumption [17, 16, 81]. The trade-off between minimizing energy cost and maximizing cloud computing services for a data center was analyzed in [82]. At the transmission grid level, receding horizon control approach [18], game theoretic approach [83, 84], and distributed control approach [85] have been developed to coordinate the operations of data centers and distributed energy resources such as renewable generation and electric vehicles.

The use of data centers to provide frequency regulation service has attracted a great deal of interest recently. Data centers need to participate in two electricity market processes to provide frequency regulation services: the hour-ahead (HA) market bidding process and the real-time operations. The existing literature can be divided into two groups based on which market/operation process was considered. In the first group of literature, the profit maximization problem of the data center is formulated to determine the optimal bidding strategy for energy and frequency regulation services. In [86], an optimization-based profit maximization problem for data centers with quality of service (QoS) constraint is formulated. The service rate is controlled to offer load reduction as an ancillary service. In [87], the problem of leveraging energy storage systems in data centers to provide frequency

regulation service and peak shaving service is studied. However, the profit maximization problem formulated in existing literature did not take the uncertainty of energy, frequency regulation service prices, and data center requests arrival rates into consideration. In addition, the financial risks associated with participating in the electricity market are not modeled. In the second group of literature, the real-time frequency regulation signal following problem of data centers is investigated. In [88, 89], the DVFS and CPU resource limit techniques are adopted to adjust the power consumption. Different real-time control policies, such as efficiency-first and priority-first policies, are proposed to follow the frequency regulation signal. Various power states of the servers are considered in [90, 91] to offer additional flexibility for power control, including active, idle, slow-to-wakeup sleep state, and shut-down power states. In [92], a stochastic dynamic programming problem is formulated to find the optimal policy for the frequency regulation service provision while reducing the quality of service degradation. In [93, 94], battery storage systems are leveraged to provide frequency regulation services. Peak demand reduction of the data center is also considered in [93]. In [95, 96], both CPU frequency and the charging schedule of electric vehicles are controlled to follow the frequency regulation signals in real-time.

In Section 3.2 we explore the ways to coordinate the operations of data centers and distributed energy resources at the electric power distribution system level. The existing work ignored the three-phase electrical wiring within a data center and modeled only balanced three-phase power systems. Moreover, we filled the knowledge gap by carefully modeling the realistic three-phase unbalanced electric power distribution network and the data center. We propose solving the distribution network phase balancing problem by shift-

ing computational loads among the servers connected to three different phase wires in a data center.

In Section 3.3, we propose a comprehensive framework for the frequency regulation provision by data centers, which covers the hour-ahead market bidding and the real-time signal following problems. A piece-wise bi-linear energy consumption model of the data center servers with default deep sleep state policy is first derived based on empirical measurements from real-world tests on servers. Dummy computing loads are introduced to control server power consumption in addition to the traditional DVFS technique. A neural network-based probabilistic model of energy and frequency regulation service prices is developed and embedded into the risk constrained optimization problem to determine the optimal energy and frequency regulation service bids for the data center in the hour-ahead market. For real-time operations, a rule-based data center power consumption control algorithm is developed, which not only enables frequency regulation signal following with high accuracy but also reduces the total response time of the requests.

3.2 Phase Balancing in Power Distribution Network with Data Center

3.2.1 Overall Framework

The overall framework of coordinating the operation of data center and DERs to reduce phase unbalance and improve operational efficiency of electric distribution networks is depicted in Fig. 3.1. The coordination framework involves interactions among three decision making entities in the DSO managed electricity market. They are the DSO, the

DERs, and the data center.

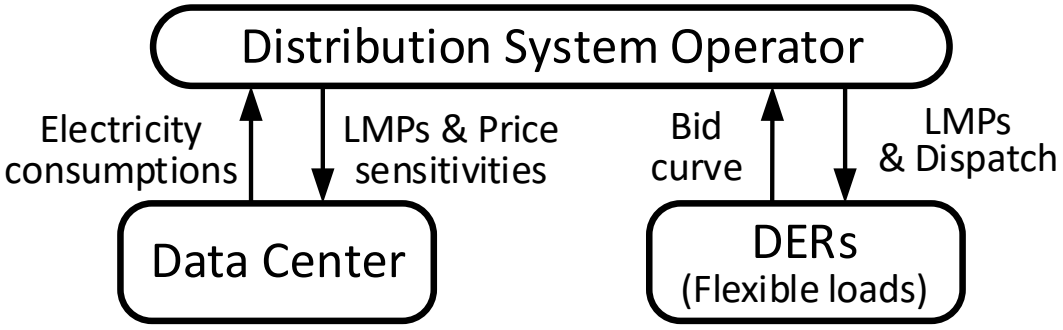

Figure 3.1: Overall coordination framework

A. Distribution System Operator

The DSO manages the distribution electricity market and adopts a transactive and iterative approach to coordinate the operations of DERs and data center. In each iteration of the market clearing process, the DSO tries to maximize the social welfare in the distribution circuit with the three-phase DC optimal power flow (DCOPF) algorithm. The inputs to the three-phase DCOPF algorithm include the price-sensitive energy bid curves from the DERs, the electricity consumption target from the data center, and forecast for fixed loads in the distribution feeder. The outputs of the three-phase DCOPF algorithm include the three-phase LMPs, the dispatch levels for the DERs, and the LMPs sensitivities. After the distribution electricity market is cleared, the DSO will send the LMPs and the dispatch operating points to the DERs, the LMPs and the prices sensitivities to the data center. The three-phase DCOPF algorithm and the derivation for three-phase LMPs sensitivities are described in Sections 3.1 and 3.2.

B. Distributed Energy Resources

The DERs proactively participate in the DSO managed distribution electricity market by submitting their single-phase price-sensitive energy bid curves on an hourly basis. The single-phase bid curves can be constructed based on the resource control model and the customers' preferences as described in [9]. If the DER is a load resource, then a price-sensitive demand bid curve will be submitted to the DSO. The demand bid curve is a graphical representation of the relationship between quantity of electricity demand and customer's willingness-to-pay. The demand bid curve must be monotonically decreasing in the price-quantity space.

C. Data Center

It is not straightforward to construct energy bid curves for data centers due to the electrical and computational coupling among the servers on the three individual phases. To illustrate this coupling effect, a simplified electrical wiring diagram of a typical data center is shown in Fig. 3.2. As shown in the figure, the computational load and electrical consumption can be partially shifted among servers connected to different phase wires by migrating computational services. However, the total electrical consumption and service level agreement constraints which link all three phases depend on the level of load shifting. Therefore, it is difficult for the data center operator to decompose its six dimensional bid curves with prices and loads of the three phases directly into three independent two-dimensional energy bid curves.

In order to enable the proactive participation of data center in distribution elec-

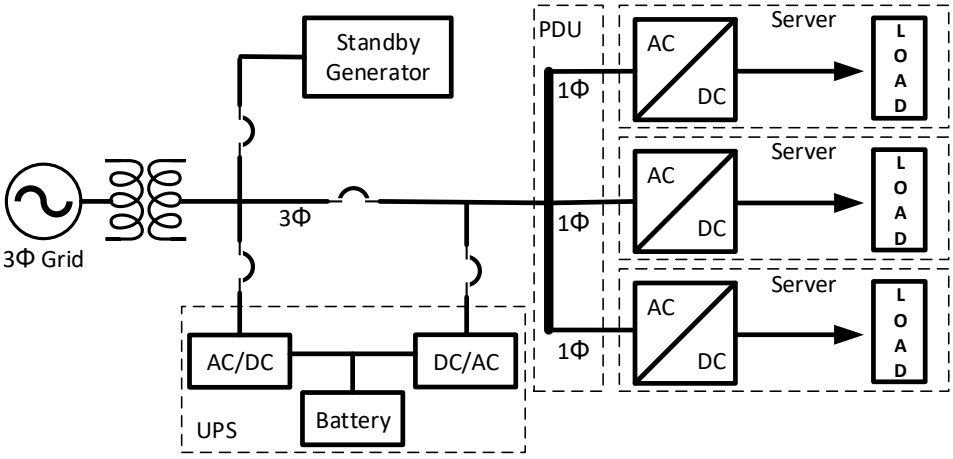

Figure 3.2: Electrical wiring diagram of a data center.

tricity market, an iterative approach is proposed to facilitate the negotiation between the DSO and the data center. In each iteration of the negotiation, the data center operator first determines the optimal load shifting plan with the latest three-phase LMPs and price sensitivities information. The data center operator then submits its electricity consumption targets for the three individual phases to the DSO. The DSO will clear the distribution electricity market and sends the updated three-phase LMPs and price sensitivities to the data center. The data center electricity cost minimization algorithm is described in Section 3.3.

3.2.2 Problem Formulation

A. Distribution Network Optimization with Three-phase DCOPF

The DSO adopts a transactive approach to coordinate the operations of DERs and data center in a distribution electricity market. To clear the distribution electricity market,

the DSO runs the three-phase DCOPF algorithm [24]. The objective of the three-phase DCOPF problem is to maximize the social welfare, which is the summation of the surplus of electricity customers and energy suppliers in a distribution system as shown in equation (3.1). The energy supplier from the transmission system is assumed to be submitting a supply offer from the reference bus to the DSO. The flexible loads are located at all other buses in the distribution network. The bid curves of DERs or flexible loads are assumed to be linear for simplicity. Hence, the customer's willingness-to-pay function at node i with phase m for electricity with the amount of ${P_f}_i^m$ is in a quadratic form. The operating constraints in the distribution system include the real power balance constraints (3.2), the distribution line thermal limit constraints (3.3), and the phase imbalance constraints (3.4).

$$\max_{P_f} \sum_{n=2}^{N} \sum_{m=1}^{3} (a_i^m ({P_f}_i^m)^2 + b_i^m {P_f}_i^m) - Cg \sum_{m=1}^{3} PG^m \tag{3.1}$$

subject to

$$PG^p - \sum_{i=2}^{N} \sum_{m=1}^{3} (DF_i^m)^p \cdot PD_i^m + P_{Loss}^p(FP) - P_{Loss}^p(FQ) = 0, p = 1, 2, 3 \tag{3.2}$$

$$|\sum_{q=2}^{N} \sum_{g=1}^{3} GSFP_{ik\_q}^{p\text{-}g} \cdot \left(-PD_q^g - E_q^g\right)| \leq PLimit_{ik}^p, \forall i, k \quad and \quad i \neq k \tag{3.3}$$

$$|\sum_{n=2}^{N} P_n^i - \sum_{n=2}^{N} P_n^j| \leq \gamma,\ i, j = 1, 2, 3\ and\ i \neq j \tag{3.4}$$

The three-phase generation shift factors $GSFP_{ik\_q}^{p\text{-}g}$, fictitious nodal demands (FNDs) E_q^g, and delivery factors (DFs) $(DF_i^m)^p$ are derived from three-phase power flow equations and three-phase admittance matrix. The three-phase GSFs, FNDs, and DFs are very different from that of the single-phase systems. These differences arise from the mutual coupling among three phases of distribution system lines. The derivation details can be found in

[24]. The iterative algorithm used to solve the FND-based three-phase DCOPF problem is described in Algorithm 3. The outputs of the iterative three-phase DCOPF algorithm include the dispatch for flexible loads, generation, the LMPs at each bus with all three phases.

The three-phase LMPs can be decomposed using the Lagrangian function of three-phase DCOPF problem. Define λ^p as the Lagrange multiplier of the constraints (2), μ_b^{p+} and μ_b^{p-} as the Lagrange multipliers of the constraints (3), and μ^{pm+} and μ^{pm-} as the Lagrange multipliers of the constraints (4). As shown in [24], the LMP of node i with phase g can be decomposed as

$$LMP_i^g = \sum_{p=1}^{3} \lambda^p (DF_i^g)^p + \sum_{b=1}^{B} \sum_{p=1}^{3} \mu_b^{p\prime} GSFP_{b\_i}^{p\_g} + \mu^{g\prime\prime} \tag{3.5}$$

Where $GSFP_{b\_i}^{p\_g}$ is generation shift factor for real power flow of the branch b with phase p when power injection is at node i with phase g. B is the set of total branches. The Lagrange multipliers $\mu_b^{p\prime}$ and $\mu^{g\prime\prime}$ are defined as

$$\mu_b^{p\prime} = \mu_b^{p+} - \mu_b^{p-}$$

$$\mu^{g\prime\prime} = \begin{cases} \mu^{12+} + \mu^{13+} - \mu_i^{12-} - \mu^{13-} & \text{if } g = 1; \\ -\mu^{12+} + \mu^{23+} + \mu_i^{12-} - \mu^{23-} & \text{if } g = 2; \\ -\mu^{13+} - \mu^{23+} + \mu_i^{13-} + \mu^{23-} & \text{if } g = 3. \end{cases}$$

$\mu_b^{p\prime}$ is the equivalent Lagrange multiplier of the thermal limit constraints (3) for branch b with phase p. $\mu^{g\prime\prime}$ is the equivalent Lagrange multiplier of the phase imbalance constraints (4) related to loading limit on phase g.

B. Three-phase LMPs Sensitivities

The sensitivities of single-phase LMPs in transmission electricity market can be derived using a perturbation approach [97]. In this section, we extend the derivation for sensitivities to three-phase LMPs in the distribution electricity market. In particular the three-phase LMPs sensitivities with respect to changes in bus demands are derived here.

Denote $h(\boldsymbol{x}, \boldsymbol{a})$ and $g(\boldsymbol{x}, \boldsymbol{a})$ as the set of equality and inequality constraints of three-phase OPF problem respectively. $\boldsymbol{x}$ represents the load and generation dispatch variables. $\boldsymbol{a}$ stands for the vector of electricity demands at all nodes.

Define z as

$$\begin{aligned} z &= f(\boldsymbol{x}, \boldsymbol{a}) \\ &= Cg \sum_{m=1}^{3} PG^m - \sum_{n=2}^{N} \sum_{m=1}^{3} (a_i^m ({P_f}_i^m)^2 + b_i^m {P_f}_i^m) \end{aligned} \tag{3.6}$$

The three-phase DCOPF can be written in compact form as

$$\min_x z = f(\boldsymbol{x}, \boldsymbol{a}) \tag{3.7}$$

subject to

$$h(\boldsymbol{x}, \boldsymbol{a}) = 0 \tag{3.8}$$

$$g(\boldsymbol{x}, \boldsymbol{a}) \leq 0 \tag{3.9}$$

By applying the perturbation technique on top of the Karush-Kuhn-Tucker first-order optimality conditions, we can obtain the sensitivities with respect to the electricity demands [98].

$$\left[d\boldsymbol{x}^T, d\boldsymbol{\lambda}^T, d\boldsymbol{\mu}^T, dz\right]^T / d\boldsymbol{a} = U^{-1}\boldsymbol{S} \tag{3.10}$$

where $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are the Lagrange multipliers vector for the equality and inequality constraints respectively. Matrix U and vector $\boldsymbol{S}$ can be derived as

$$U = \begin{bmatrix} F_x & 0 & 0 & -1 \\ F_{xx} & H_x^T & G_x^T & 0 \\ H_x & 0 & 0 & 0 \\ G_x & 0 & 0 & 0 \end{bmatrix} \tag{3.11}$$

$$\boldsymbol{S} = -\left[F_a^T, F_{xa}^T, H_a^T, G_a^T\right]^T \tag{3.12}$$

where F_x and F_a are the first order derivatives of the objective function with respect to $\boldsymbol{x}$ and $\boldsymbol{a}$. F_{xx} is the second order derivative of the Lagrange function with respect to $\boldsymbol{x}$. F_{xa} is the second order derivative of the Lagrange function with respect to $\boldsymbol{x}$ and then $\boldsymbol{a}$. H_x, H_a, G_x, and G_a are the first order derivatives of the equality and binding inequality constraints with respect to $\boldsymbol{x}$ and $\boldsymbol{a}$. The detailed derivation can be found in [98]. Taking derivatives on both sides of equation (5) with respect to fixed demand of node u phase v, we get the LMPs sensitivities of node i phase g with respect to P_u^v.

$$\frac{\partial LMP_i^g}{\partial P_u^v} = \sum_{p=1}^{3} \frac{\partial \lambda^p}{\partial P_u^v}(DF_i^g)^p + \sum_{b=1}^{B}\sum_{p=1}^{3} \frac{\partial \mu_b^{p\prime}}{\partial P_u^v} GSFP_{b\_i}^{p\_g} + \frac{\partial \mu^{g\prime\prime}}{\partial P_u^v} \tag{3.13}$$

Where the derivatives of Lagrange multipliers of non-binding inequality constraints with respect to P_u^v are zeros. Now, the derivatives of Lagrange multipliers in equation (10) can be substituted into (13) to calculate the three-phase LMPs sensitivities.

C. Data Center Electricity Cost Minimization

Since majority of the electrical appliances used in the data center cooling systems consume three-phase electrical power, they can not be leveraged to address phase balancing

problem. Hence, they are not modeled in this work. In this section, the data center electricity cost minimization problem only considers the electricity cost from the servers. Assume there are N_S^p servers on phase p. Define M^{rp} as a $N_0^r \times 1$ binary variable vector. If the i-th element of M^{rp} is 1, it means that the i-th VM is moving from a server on phase r to phase p. N_0^r denotes the number of VMs running on the servers of phase r initially.

The objective function of the data center is to minimize its electricity cost as shown in equation (3.14). The electricity costs of all servers equal to the dot product of updated LMPs vector after the VMs live migration and the vector of per phase electricity consumption of servers $\boldsymbol{P_{dc}}$. In the objective function, $\boldsymbol{LMP} = [LMP^1, LMP^2, LMP^3]^T$ denotes the LMPs for the three different phases at the data center bus and $\boldsymbol{P_{dc}} = [P_{dc}^1, P_{dc}^2, P_{dc}^3]^T$ denotes the electricity consumption from the servers on the three-phases. $\boldsymbol{P_{dc}^0}$ stands for the initial value of $\boldsymbol{P_{dc}}$. The updated LMPs vector after the VMs live migration is estimated by using the LMPs sensitivities $\partial \boldsymbol{LMP}/\partial \boldsymbol{P_{dc}}$, which is a 3×3 matrix. The LMPs sensitivities are introduced into the data center electricity cost minimization process to serve as a damping factor which prevents oscillation of computing load shifts in the data center and the DSO's negotiation process. Without the LMP sensitives, the data center will aggressively move its load from the phase with higher price to the phase with lower price without considering the impacts of the move on LMPs. This could prevent the iterative negotiation process between the DSO and the data center from reaching an equilibrium point.

The electricity consumption of a particular phase equals to the sum of electricity consumptions from each individual server connected to the phase wire in equation (3.15). The electricity consumption of each server includes a dynamic component and an idle com-

ponent as shown in equation (3.16) [82]. The dynamic component of the server electricity consumption is closely related to the server utilization rate which is modeled in equation (3.17). The utilization rate of server i on phase p, U_i^p, can be estimated based on the requests arrival rate for servers on phase p and the live migration of VMs [17]. During the migration period, computational loads increase on the servers which the VMs migrated to and from. After the migration period, computational loads increase/decrease on the servers which the VMs migrated to and from. For simplification purpose, a uniform utilization of servers is assumed for each phase.

$$\min_{M^{pr},p,r=1,2,3,p\neq r} [\boldsymbol{LMP} + \frac{\partial \boldsymbol{LMP}}{\partial \boldsymbol{P_{dc}}}(\boldsymbol{P_{dc}} - \boldsymbol{P_{dc}^0})]^T \boldsymbol{P_{dc}} \tag{3.14}$$

subject to

$$P_{dc}^p = \sum_{i=1}^{N_S^p} {P_S}_i^p,\ p = 1,2,3 \tag{3.15}$$

$${P_S}_i^p = P_{dyn_i}^p * U_i^p + P_{idlei}^p, \quad \forall i,\ p = 1,2,3 \tag{3.16}$$

$$\begin{aligned} U_i^p = \big\{ (Ar^p)^T \cdot \mathbf{1} - \sum_{r=1,r\neq p}^{3} \big[(1 - \frac{T^{pr}}{T_{int}})(Ar^p)^T \cdot M^{pr} \big] \\ \sum_{r=1,r\neq p}^{3} \big[(1 - \frac{T^{rp}}{T_{int}})(Ar^r)^T \cdot M^{rp} \big] + \sum_{r=1,r\neq p}^{3} R^{rp} M^{rp} \\ + \sum_{r=1,r\neq p}^{3} R^{pr} M^{pr} \big\} / \sum_{i=1}^{N_S^p} {R_{cap}}_i^p, \quad \forall i,\ p = 1,2,3 \end{aligned} \tag{3.17}$$

In equation (3.17), T^{pr} denotes the VM migration time from phase p to r and T_{int} is the market clearing time step. Ar^p denotes the vector of requests arrival rate of VMs on phase p before the migration. It has a dimension of $N_0^p \times 1$. ${R_{cap}}_i^p$ denotes the request

processing capability of server i on phase p. $\mathbf{1}$ is a vector of ones with the same dimension as Ar^p.

The data center electricity cost minimization formulation also includes two sets of constraints related to the service level agreement (3.18) and (3.19).

During the live migration of VMs, the response time constraints in the service level agreement is modeled as

$$T_r^p = {a_0}^p + {a_1}^p \cdot \big((Ar^p)^T \cdot \mathbf{1} + \sum_{r=1, r\neq p}^{3} R^{rp} M^{rp} + \sum_{r=1, r\neq p}^{3} R^{pr} M^{pr}\big)/N_S^p \leq T_{SLA}, \quad p = 1, 2, 3 \tag{3.18}$$

After the migration of VMs, the response time constraints in the service level agreement is modeled as

$$T_r^p = {a_0}^p + {a_1}^p \big((Ar^p)^T (\mathbf{1} - \sum_{r=1, r\neq p}^{3} M^{pr}) + \sum_{r=1, r\neq p}^{3} (Ar^r)^T M^{rp}\big)/N_S^p \leq T_{SLA}, \quad p = 1, 2, 3 \tag{3.19}$$

3.2.3 Numerical Study

A. Simulation Setup

The standard IEEE 4-bus distribution test feeder [99] is modified to validate the effectiveness of the proposed phase balancing algorithm with data center. The data center is located at node 2. The fixed loads and flexible loads are located at node 4. The transmission system is assumed to supply electric power to the distribution network through the distribution substation at a price of $\$0.6/kWh$. The total amount of fixed demands and price-sensitive demands are summarized in Table 3.1. Two simulation cases with different degree of unbalance are created. The distribution feeder is slightly unbalanced in case 1

and heavily unbalanced in case 2. The power imbalance limit between any two phases in the distribution feeder is set to be $60KW$.

The price-sensitive demand bid curves of flexible loads on the three phases are assumed to be linear functions as $Price^1 = 1 - P_f^1/200$, $Price^2 = 1 - P_f^2/250$, and $Price^3 = 1 - P_f^3/300$. $Price^1$, $Price^2$, and $Price^3$ are the bidding prices for the three phases. The price ranges of the three demand bid curves are from $\$0.1/kWh$ to

Table 3.1: Fixed and flexible load profile

Node 4		Phase A	Phase B	Phase C
Fixed Load Capacity (KW)	Case 1	460	500	530
	Case 2	420	500	580
Flexible Load Capacity (KW)		180	225	270

In the simulation, the data center powers 400 servers on each of the three phase wires through a PDU. The maximum power rating of each server is assumed to be $500W$. The maximum dynamic power and idle power of each server are $400W$ and $100W$ respectively. The data center is operating in a homogeneous computing environment. It is assumed that a total of 1200 VMs are running on the servers and they are distributed uniformly on all three phases. Each VM processes 200 requests per second. It is also assumed that the servers on each phase can host up to 800 VMs. Live migration is implemented in the data center where a VM is moved from a server on one of the phases to another server on the other phase without the need to bring down the VM instance. It is assumed that live migration can be finished within 10 minutes. The computational cost of live migration of each VM is assumed to be equivalent to the processing time for 24 request/s. The average response

time of each server is a linear function with respect to the number of requests as shown in equations (3.18) and (3.19) with parameters $a_0 = 0.2s$ and $a_1 = 8 \times 10^{-4} s/request$. The upper limit of response time is set at $500ms$ according to the service level agreement.

B. Simulation Results

The LMPs and electricity consumptions on all three phases of the data center are calculated with and without phase balancing for the two different unbalance cases.

Table 3.2: LMPs and data center electricity consumption

Case	Phase balancing	Price & Power	Phase A	Phase B	Phase C
1	Without	($/KWh)	0.5048	0.6015	0.6988
		(KW)	120.0	120.0	120.0
	With	($/KWh)	0.5454	0.6015	0.6582
		(KW)	130.5	120.0	110.0
2	Without	($/KWh)	0.3241	0.6015	0.8795
		(KW)	120.0	120.0	120.0
	With	$(/KWh)	0.4543	0.6015	0.7494
		(KW)	153.8	120.0	87.9

As shown in Table 3.2, in both cases electricity load shifts from servers on phase c which has higher price to phase a which has lower price. After phase balancing, the degree of unbalance of the distribution feeder is reduced which leads to smaller price difference between phase a and phase c. The amount of load shift and reduction in price difference is higher in the heavily unbalanced case than the slightly unbalanced case.The electricity cost of data center and total surpluses are reported in Table 3.3. As shown in the table,

the phase balancing algorithm not only reduces the electricity bill of the data center but also increases the total surpluses of the flexible loads and the supplier. The savings in the heavily unbalanced case is much more significant than the slightly unbalanced case. In case 2, the phase balancing algorithm reduces the electricity bill of the data center by more than 4.0% and increases the total surpluses by 35%. Note that the savings also depend on the price elasticity of demand.

Table 3.3: Electricity cost and total surpluses

	Case	Without Phase Balancing	With Phase Balancing
Electricity cost of data center ($)	1	216.6	215.8
	2	216.6	207.9
Total Surpluses ($)	1	57.3	58.9
	2	39.9	53.9

Instead of keeping the phase imbalance limit γ in equation (3.4) constant at 60 KW, we try to reduce the phase imbalance as much as possible without making the savings of the data center and the flexible loads worse. The simulation results show that the percentage reduction in phase imbalance varies with the electric power rating of the data center. In the simulation setup of case 2, the power rating of the data center is about 25% of the total feeder demand. In this case, our proposed algorithm can reduce the phase imbalance by 100%. In other words, by shifting computational loads in the data center, the electric loads on the distribution feeder can be completely balanced. By gradually reducing the size of the data center by 40%, 45%, and 50%, the reduction in phase imbalance also decreases to about 93%, 83%, and 73%.

3.2.4 Conclusions

This work develops an iterative scheme to coordinate the operations of data center and DERs to tackle the electric distribution network phase balancing problem. We also derived the three-phase LMPs sensitivities in a distribution electricity market and integrated the price sensitivities into the data center's electricity cost minimization algorithm as damp factors. Comprehensive simulations are conducted on a modified IEEE distribution test feeder to demonstrate the effectiveness of our proposed phase balancing algorithm. The simulation results showed the degree of unbalanced of a distribution feeder is significantly decreased by up to 100%, and the electricity cost of a data center is reduced by more than 4.0%.

3.3 Frequency Regulation Service Provision in Data Center with Computational Flexibility

3.3.1 Overall Framework

The overall framework of the frequency regulation service provision by a data center is depicted in Fig. 3.3. The overall framework involves interactions between a transmission system operator (TSO) and a data center (DC) in two electricity market processes: hour-ahead market and real-time operations. The details of the frequency regulation service provision framework is described in the next three subsections. The proposed framework is applicable to different electricity markets. The specific implementation of the bidding strategy can be easily adjusted to different market rules.

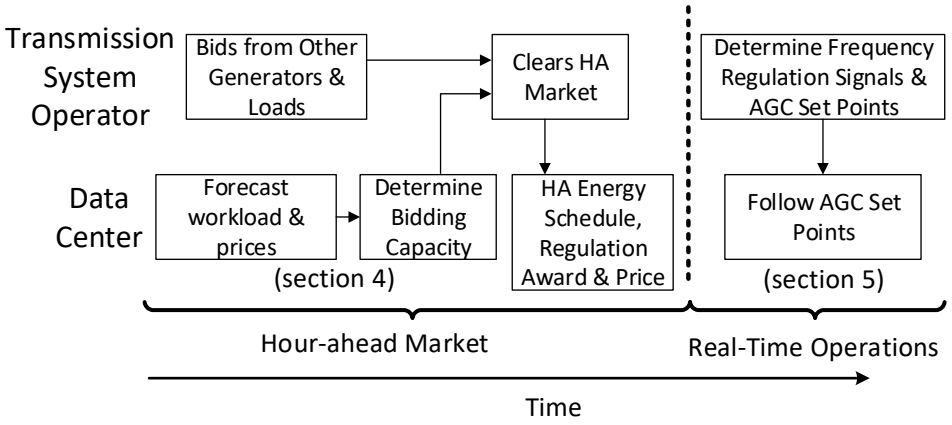

Figure 3.3: Overall framework of frequency regulation service provision by a data center

A. Data Center's Participation in Electricity Market

As shown in Fig. 3.3, in order to provide frequency regulation services, the data center is required to participate in two electricity market processes: HA market and real-time operations.

In the HA market, the data center will first predict the prices for energy and frequency regulation services, and the workload of the data center for the next operating hour. The data center will then determine the optimal bidding capacity for energy and frequency regulation services which maximize its expected net benefits subject to certain risk limits. After the HA market is cleared by the transmission system operator, the data center will receive the hour-ahead energy schedule, the award for frequency regulation service, and the cleared prices for energy and frequency regulation.

In real-time operations, the data center receives the frequency regulation signals and automatic generation control (AGC) set points from the transmission system operator every 2 seconds. The frequency regulation signals range from -1 to 1. The signals are

negative (positive) when the system requests frequency regulation down (up) services. The AGC set points specify the amount of load the data center should consume. The AGC set points are equal to the summation of the HA market energy schedule plus the product of the frequency regulation signals and frequency regulation service awards. Upon receiving the AGC set points, the data center adjusts its energy consumption to follow the set points. Data centers can accurately follow the AGC set points by dynamically routing arriving requests to various servers, changing the operating frequency of CPUs and inserting dummy loads at the server level.

The physical and contractual constraints of the data center need to be taken into consideration when participating in the electricity market. First, the bidding capacity for energy P_{base} and frequency regulation service B_{cap} should be determined in such a way that the maximum and minimum power consumption limits P_{max} and P_{min} of the data center will not be violated. If the submitted bids are accepted, then in real-time operations the AGC set points for the data center ranges from $P_{base} - B_{cap}$ to $P_{base} + B_{cap}$. The data center needs to make sure $P_{base} + B_{cap} \leq P_{max}$ and $P_{base} - B_{cap} \geq P_{min}$. Second, as a cloud computing service provider, the data center also needs to satisfy the service level agreement (SLA) and maintain the QoS. Hence, the control of request routing, CPU frequency, and dummy loads are limited by the SLA requirements.

Finally, note that in electricity markets such as Pennsylvania-New Jersey-Maryland Interconnection (PJM), there are two types of frequency regulation services, $RegA$ and $RegD$. The real-time regulation signal of $RegD$ service is much more volatile than that of $RegA$ service, and the price of $RegD$ service is higher than that of $RegA$ service. The data

center is capable of controlling its server energy usage in real-time to follow the volatile *RegD* service signals. Hence, it is suitable for the data center to provide such premium frequency regulation services and receive higher compensation from the electricity market. Fig. 3.4 shows an example of daily prices for frequency regulation services and energy in the PJM market. For about 32% of hours in year 2017 and 2018, the frequency regulation price of RegD service is higher than the energy price in PJM market. As the penetration level of renewable energy increases, the demand for frequency regulation service will rise as well. This will further increase the percentage of hours where the frequency regulation service price is higher than the energy price.

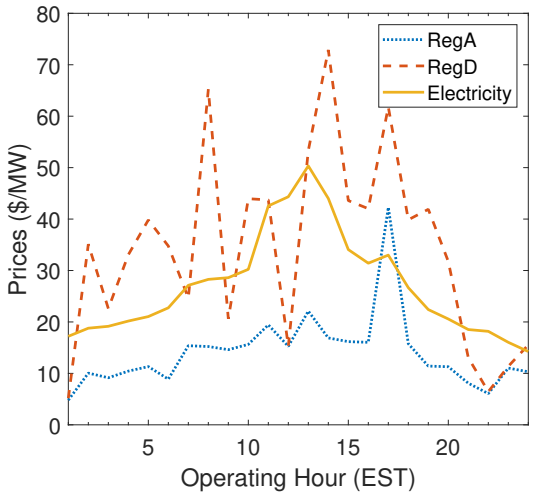

Figure 3.4: Prices for frequency regulation services and energy in PJM market

B. Transmission System Operator

In the hour-ahead market process, the transmission system operator first receives both energy and frequency regulation service bids from generators and data centers. It

then clears the hour-ahead market to determine the hour-ahead energy schedule and prices for energy and frequency regulation services. The objective is to minimize the total energy and frequency regulation service costs while satisfying the electric loads [100]. The market clearing results will be sent to data centers and other market participants. In the real-time operations, the transmission system operator will first measure area control error and compute the frequency regulation signals of the system aiming to reduce the area control error to zero in a distributed fashion[101]. The individual generator and data center's AGC set points will be calculated based on the frequency regulation signal, hour-ahead energy schedule, and frequency regulation service awards. The updated AGC set points will be sent to the generators and data centers every 2 seconds.

C. Performance-based Compensation

The final compensation for providing frequency regulation services depends on both the frequency regulation service award amount and the real-time AGC set point signal following performance. The signal following performance is quantified by the performance score in PJM market [102]. It consists of three components: accuracy, delay, and precision.

The accuracy score is the correlation between the AGC set point signals and data center's response. It is calculated over a five-minute period with 10-second granularity. The calculation is performed repeatedly with 10-second delays propagated over five minutes, where the best score is used. The delay score is based on the time delay between the control signal and the point of the highest correlation. The delay score will be 100% if the best correlation is at 0 or 10-second delay. It decreases as the delay time increases until the 5-minute mark. The precision score is calculated based the instantaneous error between the

control signal and the regulating unit's response. The final performance score is the average of the three components.

D. Energy Model of Server

To build an energy model of a server, we empirically profiled a server running CentOS 7 with 32 Intel Xeon E5 cores across different working frequencies between 1.2GHz and 2.1GHz (maximum frequency without turbo-boosting) with the default C6 sleep policy. We focus the energy model on processors because processors are the largest consumers of power in data centers and have the largest dynamic range [103]. In addition, it has been widely observed that processor power consumption can be used as a proxy for whole-server power [103, 104, 105].

As a workload, we used the Web Search benchmark from Cloudsuite [106] with a ramp time and steady state time of 30 and 250 seconds respectively. The measurements taken from the tests include the average power consumption and percentage of idle time. To find the maximum computing load a server can handle, we gradually increase the number of clients in the benchmark at every frequency until it fails to satisfy QoS and pick the greatest value. The idleness and power measurements are performed using *powertop* and *rapl-read* [107] respectively, while the frequency is scaled using *cpufreq* drivers.

The percentage of C6 sleep time and CPU power consumption of a single server with different request rates (per second) are depicted in Fig. 3.5. As shown in the figure, the default C6 sleep time percentage decreases with the increase of request per second (RPS) almost linearly for each frequency. The C6 sleep time of a server running at maximum frequency f_{max}, i.e. 2.1 GHz, reaches zero at 1230 RPS, which is deemed as the maximum

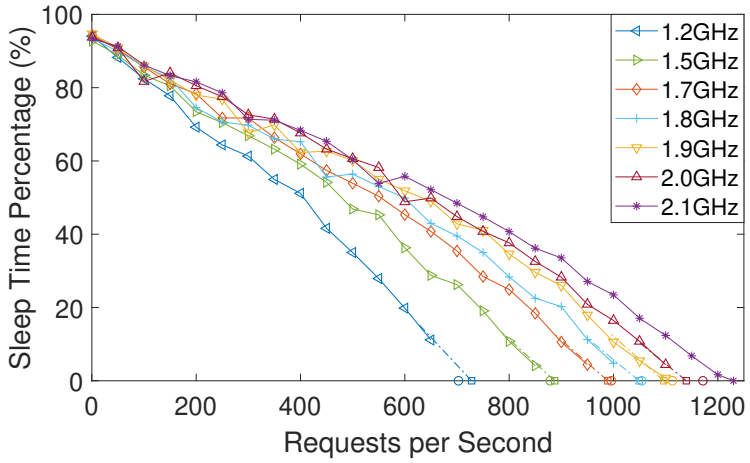

Figure 3.5: C6 sleep time versus request per second rate

capacity of the server $cap_{max}(f_{max})$. Fig. 3.5 also shows the maximum capacity of the server under various frequencies $cap_{max}(f)$ increases approximately linearly with the frequency of the server CPU.

Hence, we can estimate the maximum capacity of the server under frequency f with:

$$cap_{max}(f) = \frac{f}{f_{max}} cap_{max}(f_{max}) \tag{3.20}$$

In Fig. 3.5, the circles on the horizontal axis represent the estimated maximum capacity under different frequencies by scaling cap_{max} with the above linear equation. The squares on the horizontal axis of the figure are the approximated maximum capacity by linear extrapolation with the last 4 points of each curve. The short distances between the circles and squares show that the linear approximation for the maximum server capacity is fairly accurate.

The utilization rate of the server i at time t is defined as the ratio of the number

of requests per second to the maximum server capacity under a particular frequency:

$$ur_i(t) = \frac{r_i(t)}{cap_{max}(f_i(t))} \tag{3.21}$$

The energy consumption model with the default sleep policy is the baseline considered in this section. At the server level, the energy consumption can be controlled by adjusting the CPU frequency and introducing the dummy computing load. The dummy load can be trivially injected by running a process that stresses the CPU with mainly compute instructions, limiting performance interference with other processes running in the server. It increases the equivalent utilization rate and decreases the sleep time percentage. The total equivalent utilization rate with the dummy load is

$$u_i(t) = ur_i(t) + ud_i(t) \tag{3.22}$$

The power consumption of a single server with the default sleep policy at different frequencies and utilization rates are depicted by the markers in Fig. 3.6. The relationship between the power consumption and utilization rate of the servers can be described by a piece-wise bi-linear function. Note that the slope of the power curve segment where the utilization rate is between 0 and 0.1 is larger than that of the segment where the utilization rate is between 0.1 and 1. It can also be observed that for a fixed utilization rate, the power consumption increases faster when the frequency increases from 2.0 GHz to 2.1 GHz than when the frequency increases from 1.2 GHz to 2.0 GHz.

Therefore, the server power can be approximated as a piece-wise bi-linear function of frequency and utilization rate as:

$$P_i(t) = \alpha_1^j f_i(t) u_i(t) + \alpha_2^j u_i(t) + \alpha_3^j f_i(t) + \alpha_4^j \tag{3.23}$$

where the four different ranges of utilization rate and frequency are defined as follows:

$$j = \begin{cases} 1, & 0 <= u_i(t) <= 0.1, 1.2 <= f_i(t) <= 2.0 \\ 2, & 0 <= u_i(t) <= 0.1, 2.0 <= f_i(t) <= 2.1 \\ 3, & 0.1 <= u_i(t) <= 1, 1.2 <= f_i(t) <= 2.0 \\ 4, & 0.1 <= u_i(t) <= 1, 2.0 <= f_i(t) <= 2.1 \end{cases} \tag{3.24}$$

The power consumption curves fitted with least square regression according to Equation (3.23) are depicted in Fig. 3.6. As shown in the figure, the piece-wise bi-linear function is well suited to model the server power consumption.

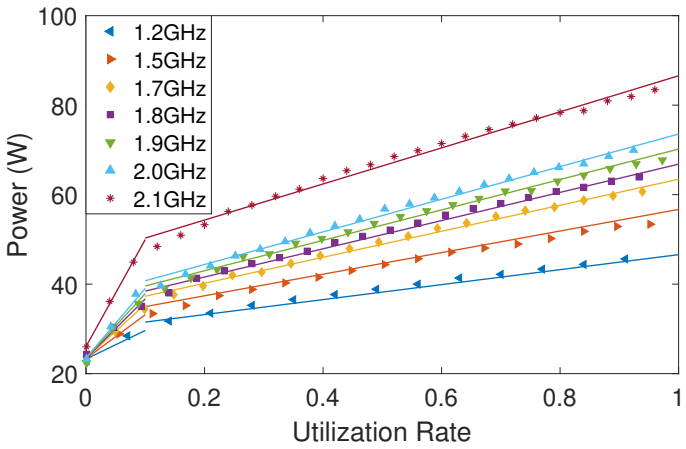

Figure 3.6: Fitted power consumption curves with default sleep policy

This piece-wise bi-linear model is typical of modern processors. Leakage energy dominates at low utilization levels (below 10%), and idle power management policies, such as sleep states and circuit-level power gating, have a profound effect. Furthermore, at 2.1GHz we observed that power increases at a faster rate. This is also typical of modern

processors that utilize frequency boosting techniques, such as Intel TurboBoost or AMD Turbo Core [108]. These policies aim to maximize performance and leverage the processor's thermal headroom. Therefore, operating at these frequencies sacrifices energy efficiency for performance, thus the increase in power consumption rate.

3.3.2 Hour-Ahead Market Frequency Regulation and Energy Bidding Strategy for Data Center

A. Problem Formulation

In the hour-ahead market, the objective of the data center is to determine the optimal bidding strategy which maximizes the expected net earnings for each hour t. The net earnings of the data center can be calculated as the difference between the revenue received from frequency regulation service provision and the total electricity cost as shown in Equation (3.25):

$$\max\ E[C_{reg}(t)score(t)B_{cap}(t) - (1+\eta_{cool})C_e(t)P_{DC}^{base}(t)] \tag{3.25}$$

where $C_e(t)$ and $C_{reg}(t)$ are electricity price and frequency regulation price for hour t. The cooling cost coefficient, i.e., ratio of cooling power over server load is denoted by $\eta_{cool} \in R^+$. The decision variables are the bidding capacity for frequency regulation service $B_{cap}(t)$ and energy consumption $P_{DC}^{base}(t)$. Note that the data center is assumed to be a price-taker in the electricity market participation process. In addition, only the power consumption from the servers are considered here.

The constraints for the net earnings maximization problem are as follows:

$$P_{DC}^{min}(t) \leq P_{DC}^{base}(t) - B_{cap}(t) \tag{3.26}$$

$$B_{cap}(t) + P_{DC}^{base}(t) \leq P_{DC}^{max} \tag{3.27}$$

$$f_{risk}(C_{reg}(t), C_{efe}(t), r(t), B_{cap}(t), P_{DC}^{base}(t)) \leq \delta_{risk} \tag{3.28}$$

where $C_{efe}(t) = (1+\eta_{cool})C_e(t)$ is the effective energy price considering cooling cost. Equations (3.26) and (3.27) represent the upper and lower bidding capacity constraints. Equation (3.28) represents the risk limit constraint for the data center's bidding strategy, where $r(t)$ represents the average request arrival rate during hour t.

If the data center does not provide frequency regulation service, then its optimal bidding strategy aims at minimizing energy cost. Hence, the risk of the joint energy and frequency regulation service bidding strategy can be defined as the expectation of the bidding strategy loss compared to the power consumption minimization scenario. Note that losses arise in cases when the revenue received from frequency regulation service provision is less than the increased energy cost:

$$\begin{aligned}
&f_{risk}(C_{reg}(t), C_{efe}(t), r(t), B_{cap}(t), P_{DC}^{base}(t)) \\
&= -\iint_V Pr(C_{reg}(t), C_{efe}(t))\big\{[C_{reg}(t)score(t)B_{cap}(t)- \\
&C_{efe}(t)P_{DC}^{base}(t)] - (-C_{efe}(t)P_{DC}^{min}(t))\big\}dC_{reg}(t)dC_{efe}(t) \\
&= \iint_V Pr(C_{reg}(t), C_{efe}(t))\big\{C_{efe}(t)[P_{DC}^{base}(t) - P_{DC}^{min}(t)] \\
&- C_{reg}(t)score(t)B_{cap}(t)\big\}dC_{reg}(t)dC_{efe}(t)
\end{aligned} \tag{3.29}$$

where V is defined as follows:

$$\begin{aligned}
V = \{C_{reg}(t), C_{efe}(t) | &C_{efe}(t)[P_{DC}^{base}(t) - P_{DC}^{min}(t)] \\
&> C_{reg}(t)score(t)B_{cap}(t)\}
\end{aligned} \tag{3.30}$$

$Pr(C_{reg}(t), C_{efe}(t))$ denotes the joint probability distribution of frequency regulation service price $C_{reg}(t)$ and effective energy price $C_{efe}(t)$.

In Equation (3.27), P_{DC}^{max} denotes the maximum power consumption of the data center, which can be calculated by summing up individual servers' power consumption at full utilization rate and maximum CPU frequency as $P_{DC}^{max} = \sum_{i=1}^{N} P_i(f = 2.1, u = 1.0)$.

In Equation (3.26) and (3.29), the minimum power consumption of data center P_{DC}^{min}, can be found by solving the following optimization problem with $r_i(t)$, $u_i(t)$, and $f_i(t)$ as decision variables:

$$P_{DC}^{min}(t) = \min \sum_{i=1}^{N} P_i(t) \tag{3.31}$$

s.t.

$$\sum_{i=1}^{N} r_i(t) = E[r(t)] \tag{3.32}$$

$$P_i(t) = \alpha_1^j f_i(t) u_i(t) + \alpha_2^j u_i(t) + \alpha_3^j f_i(t) + \alpha_4^j \tag{3.33}$$

$$u_i(t) = ur_i(t) \leq 100\% \tag{3.34}$$

$$ur_i(t) = \frac{r_i(t)}{cap_{max}(f_i(t))} \tag{3.35}$$

$$rt_i(t) = f_{rt}(f_i(t), u_i(t)) \leq rt_{SLA} \tag{3.36}$$

The objective function (3.31) aims at minimizing the summation of the power consumption of each server. Equations (3.32) to (3.36) represent the operation constraints of the data center. We assume a homogeneous computing environment in the data center. Hence, Equation (3.32) ensures that the summation of requests routed to each server should be equal to the total requests received by the data center. Equation (3.33) represents the power consumption model of each server where only the CPU power is considered. As shown

in Section 3.3.1, the power consumption of a server with the default sleep state policy is a piece-wise bi-linear function of CPU frequency and utilization rate. Equation (3.34) enforces the upper limit of the CPU utilization rate. Note that in the power consumption minimization problem, the dummy computing load must be zero. Hence $u_i(t) = ur_i(t)$. Equation (3.36) represents the service level agreement constraint which sets upper limits on the response time of the 90th percentile of the requests. As shown in Fig. 3.7, the

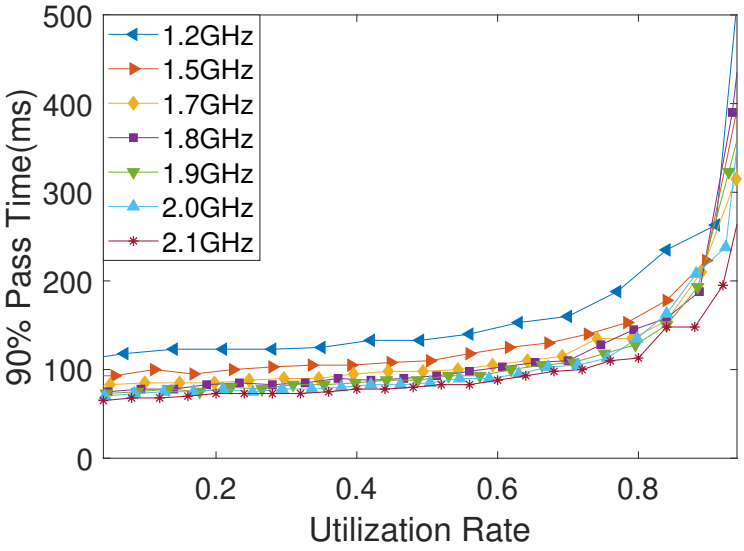

Figure 3.7: 90% pass time versus utilization rate

response time is a function of utilization rate. Hence, we can set an equivalent upper bound on the utilization rate $u_i(t)$. For example, the utilization rate limit of $rt_{SLA} = 115ms$ at $f = 2.1GHz$ is about 0.8. Define $u_{max}(f)$ as the corresponding utilization limit of the SLA. Although the explicit functional form of the utilization rate limit is not modeled here, the estimation for the minimum power consumption P_{DC}^{min} can still be performed as shown in the 3.3.6.

B. Problem Reformulation

The energy price C_e is rarely negative in practice. In PJM market, there are only 24 hours in total with negative energy prices during year 2017 and 2018. When the energy price becomes negative, the net earnings of a data center when providing both frequency regulation and energy services are greater than when the data center tries to minimize power consumption. Hence, we only consider the cases when $C_{efe} > 0$.

The Lagrange function of the optimization problem (3.25) to (3.28) is

$$\mathcal{L} = E[C_{efe}(t)]P_{DC}^{base}(t) - E[C_{reg}(t)score(t)]B_{cap}(t) + \mu_1(P_{DC}^{min}(t) + B_{cap}(t) - P_{DC}^{base}((t))$$
$$+ \mu_2(B_{cap}(t) + P_{DC}^{base}((t) - P_{DC}^{max}(t)) + \mu_3(f_{risk} - \delta) \tag{3.37}$$

where $\mu_1 \geq 0$, $\mu_2 \geq 0$, and $\mu_3 \geq 0$ are the corresponding Lagrange multipliers. By taking partial derivative of the Lagrange function with respect to $P_{DC}^{base}(t)$, we obtain

$$\frac{\partial \mathcal{L}}{\partial P_{DC}^{base}(t)} = E[C_{efe}(t)] - \mu_1 + \mu_2 + \mu_3 \frac{\partial f_{risk}}{\partial P_{DC}^{base}(t)} \tag{3.38}$$

As shown in 3.3.6, $(\partial f_{risk})/(\partial P_{DC}^{base}(t)) \geq 0$. Intuitively, the risk of providing frequency regulation service increases with $P_{DC}^{base}(t)$ because the electricity cost increases with $P_{DC}^{base}(t)$.

At the optimum point, (3.38) is equal to zero. Hence, we have

$$E[C_{efe}(t)] = \mu_1 - \mu_2 - \mu_3 \frac{\partial f_{risk}}{\partial P_{DC}^{base}(t)} > 0 \tag{3.39}$$

Therefore, at optimal solutions,

$$\mu_1 > \mu_2 + \mu_3 \frac{\partial f_{risk}}{\partial P_{DC}^{base}(t)} \geq 0 \tag{3.40}$$

Hence, the constraint (3.26) is binding at the optimum point, i.e.,

$$P_{DC}^{base}(t) = B_{cap}(t) + P_{DC}^{min}(t) \tag{3.41}$$

Therefore, the objective function (3.25) can be reformulated as:

$$\begin{aligned} & E[C_{reg}(t)score(t)B_{cap}(t) - C_{efe}(t)P_{DC}^{base}(t)] \\ & = E[C_{reg}(t)score(t) - C_{efe}(t)]B_{cap}(t) \\ & - E[C_{efe}(t)]P_{DC}^{min}(t) \end{aligned} \tag{3.42}$$

Let's define the price difference $C_{dif}(t)$ as $C_{dif}(t) = C_{reg}(t)score(t) - C_{efe}(t)$, and the expectation of $C_{dif}(t)$ as $\overline{C}_{dif}(t) = E[C_{dif}(t)]$. If the estimator of $r(t)$ is unbiased, then $P_{DC}^{min}(t)$ calculated based on $r(t)$ is also unbiased. Note that the second term on the right hand side of equation (3.42) is not a function of the decision variable $B_{cap}(t)$. Hence, maximizing the objective function (3.42) is equivalent to maximizing $\overline{C}_{dif}(t)B_{cap}(t)$, i.e., the extra benefits of providing frequency regulation services compared to the minimum power consumption strategy.

Similarly, by leveraging the equality (3.41), the risk limit constraint (3.28) can be simplified as

$$-\int_{-\infty}^{0} Pr(C_{dif}(t))C_{dif}(t)B_{cap}(t)dC_{dif}(t) \leq \delta_{risk} \tag{3.43}$$

In summary, the optimization problem (3.25) to (3.28) can be reformulated as:

$$\max \overline{C}_{dif}(t)B_{cap}(t) \tag{3.44}$$

s.t.

$$-\int_{-\infty}^{0} Pr(C_{dif}(t))C_{dif}(t)B_{cap}(t)dC_{dif}(t) \leq \delta_{risk} \tag{3.45}$$

$$0 \leq B_{cap}(t) \leq \frac{P_{DC}^{max}(t) - P_{DC}^{min}(t)}{2} \tag{3.46}$$

C. Solution Methodology

In order to solve the optimization problem (3.44)-(3.46), the probability distribution of the price difference C_{dif} needs to be modeled and estimated first.

A feed-forward neural network can be trained to estimate the probability distribution of C_{dif} based on the observed price differences and input features X such as the historical prices, load and generation information. The conditional distribution of price difference given observed features and trained neural network parameters is assumed to be Gaussian.

$$Pr(C_{dif}|X, W) = \mathcal{N}(C_{dif}|y(X, W), \beta) \tag{3.47}$$

Where X denotes the input features, W denotes the weights of the neural network, and β denotes the variance of the Gaussian noise. $y(X, W)$ denotes the output of the neural network, which is the mean value of price difference variable which follows the Gaussian distribution.

Given a data set of N_d independent, identically distributed observations along with corresponding target values for price differences $\{(x_i, C^i_{dif}), i = 1, 2..., N_d\}$, we can construct the corresponding negative logarithm of the likelihood function:

$$\frac{1}{2\beta}\sum_{i=1}^{N_d}(y(x_i, W) - C^i_{dif})^2 - \frac{N_d}{2}\ln\frac{1}{\beta} + \frac{N_d}{2}\ln(2\pi) \tag{3.48}$$

The weights of the neural network W can be obtained by maximizing the likelihood or minimizing the sum-of-square error function given by

$$\sum_{i=1}^{N_d}(y(X, W) - C^i_{dif})^2 \tag{3.49}$$

Denote the W obtained by minimizing the sum-of-square error as W_{ml}. By making the partial derivative of Equation (3.48) with respect to β equal to zero, β_{ml} can be obtained as

$$\beta_{ml} = \frac{\sum_{i=1}^{N_d}(y(x_i, W_{ml}) - C_{dif}^i)^2}{N_d} \tag{3.50}$$

After training the neural network and obtaining the network parameters W_{ml} and β_{ml}, we have $\overline{C}_{dif}(t) = y(x(t), W_{ml})$, and

$$Pr(C_{dif}(t)|x(t), W_{ml}) = \mathcal{N}(C_{dif}(t)|\overline{C}_{dif}(t), \beta_{ml}). \tag{3.51}$$

Therefore, the closed form solution of optimization problem (3.44)-(3.46) is

When $\overline{C}_{dif}(t) > 0$,

$$B_{cap}(t) = min\left\{ \frac{P_{DC}^{max}(t) - P_{DC}^{min}(t)}{2}, \frac{-\delta_{risk}}{\int_{-\infty}^{0} \frac{1}{\sqrt{2\pi\beta_{ml}}} e^{-\frac{(C_{dif}(t)-\overline{C}_{dif}(t))^2}{2\beta_{ml}}} C_{dif}(t) dC_{dif}(t)} \right\} \tag{3.52}$$

Otherwise $B_{cap}(t) = 0$. Therefore, the actual gain is influenced by the accuracy of the price difference forecast model as the $B_{cap}(t)$ is determined by $\overline{C}_{dif}$ and β_{ml}.

Note that to obtain $P_{DC}^{min}(t)$ in (3.52), the requests arrival rate needs to be modeled and estimated. In this section, we adopt the auto-regressive integrated moving average (ARIMA) [109] model to approximate the time series of requests arrival rate.

3.3.3 Following Real-Time Frequency Regulation Signal

A. Problem Formulation

In the real-time frequency regulation signal following stage, the goal of the data center is to minimize the total response time of the requests while following the frequency

regulation signals accurately. Therefore, the task of following real-time frequency regulation signal is equivalent to solving the following optimization problem:

$$\min \sum_{i=1}^{N} r_i(t) rt_i(t) \tag{3.53}$$

s.t.

$$P_{set}(t) - P_{DC}(t) = 0 \tag{3.54}$$

where $P_{set}(t) \triangleq P_{DC}^{base}(t) + B_{cap}(t) RegD(t)$ is the AGC set point sent to the data center by the transmission system operator. $P_{DC}^{base}(t)$ is the energy dispatch level of the data center and $B_{cap}(t) RegD(t)$ is the amount of frequency regulation service the data center is required to provide. Note that $RegD(t)$ is the frequency regulation signal which ranges from -1 to 1. $rt_i(t)$ is the average request response time of server i. The response time of each request is determined by the utilization rate of each server. At last, $P_{DC}(t) \triangleq \sum_{n=1}^{N} P_i(t)$, where $P_i(t)$ can be calculated by Equation (3.33). The decision variables of the optimization problem include CPU frequency (discrete variable) and the dummy load (continuous variable) of each server.

B. Rule-Based Data Center Power Consumption Control Algorithm

In order to make online adjustments of total power consumption as the frequency regulation signal is updated every 2 seconds, a rule-based control strategy is proposed.

The uniform server utilization rate at time t is defined by routing requests evenly to all servers operating with a CPU frequency of 2.1 GHz as

$$u_{uni}(t) = \frac{r(t)/N}{cap_{max}(f = 2.1)} \tag{3.55}$$

The total power consumption at time t by uniformly routing requests to all servers running at 2.1 GHz can be obtained as

$$P_{uni}(t) = \sum_{i}^{N} P_i(f = 2.1, u = u_{uni}(t)) \tag{3.56}$$

In order to accurately follow the frequency regulation signals while minimizing the total request response time, the operating strategy of the data center varies according to the AGC set point and total number of requests received by the data center as follows:

1. $P_{set}(t) \geq P_{uni}(t)$;

2. $P_{set}(t) < P_{uni}(t)$ and $r(t) \leq u_{max}(2.0) \times N \times cap_{max}(f = 2.0)$

3. $P_{set}(t) < P_{uni}(t)$ and $r(t) > u_{max}(2.0) \times N \times cap_{max}(f = 2.0)$

The operating strategy of the data center under each of the three cases will be presented in detail.

CASE 1: When $P_{set}(t) \geq P_{uni}(t)$, the minimum request response time can be achieved by uniformly routing requests to all the servers and adding dummy loads until the AGC set point is met. A proof for why the proposed data center operating strategy achieves minimum request response time while accurately following the frequency regulation signal is provided in the 3.3.6.

To follow the frequency regulation signals accurately, dummy computing loads need to be added to increase the server utilization rate. The amount of dummy load $u_d(t)$ needed can be calculated as follows:

$$u_d(t) = u(t) - u_{uni}(t) \tag{3.57}$$

where $u(t)$ can be found by solving:

$$P_{set}(t) = \sum_{i=1}^{N} P_i(f = 2.1, u(t)) = N(2.1\alpha_1^j u(t) + \alpha_2^j u(t) + 2.1\alpha_3^j + \alpha_4^j) \tag{3.58}$$

Hence, the closed form solution of $u(t)$ is as follows:

$$u(t) = \frac{P_{set}(t) - N(2.1\alpha_3^j + \alpha_4^j)}{N(2.1\alpha_1^j + \alpha_2^j)} \tag{3.59}$$

Where

$$j = \begin{cases} 2, & P_{set}(t)/N < P_i(f = 2.1, u = 0.1) \\ 4, & P_{set}(t)/N \geq P_i(f = 2.1, u = 0.1) \end{cases} \tag{3.60}$$

CASE 2: When $P_{set}(t) < P_{uni}(t)$ and $r(t) \leq u_{max}(2.0) \times N \times cap_{max}(f = 2.0)$, the data center's operating strategy works as follows. Note that in this case the power consumption needs to be reduced from $P_{uni}(t)$ to $P_{set}(t)$ and the SLA can be satisfied with all servers running at 2.0 GHz. We will start from the baseline operating strategy where the requests are uniformly routed to all servers running at $f = 2.1$ GHz. Then we select n servers whose requests are packed to n' servers running at 2.0 GHz with the maximum utilization rate $u_{max}(2.0)$ which does not violate the SLA. The remaining $n-n'$ servers are kept in idle state. By carefully choosing n and n', the data center is capable of closely following the frequency regulation signals. Although the proposed data center operating strategy increased the response time for some of requests compared to the uniform routing benchmark, it minimizes the number of requests with increased response time as shown in 3.3.6.

The data center control parameters n and n' can be calculated online as follows.

The total energy consumption of the data center in CASE 2 include energy consumption from n' servers with packed requests, $n - n'$ servers in idle, and $N - n$ servers operating under the uniform routing and the maximum frequency.

$$P_{DC}(t) = \sum_{i=1}^{n'} P_i(f = 2.0, u = u_{max}(2.0)) + \sum_{i=n'+1}^{n} P_0 + \sum_{i=n+1}^{N} P_i(f = 2.1, u = u_{uni}(t)) \tag{3.61}$$

The number of servers with packed requests, n' can be expressed as a function of n:

$$n' = \left\lceil \frac{nr(t)}{u_{max}(2.0)Ncap_{max}(f = 2.0)} \right\rceil \leq n \tag{3.62}$$

By setting $P_{set}(t) = P_{DC}(t)$ and substituting (3.62) into (3.61), we can solve for n as follows:

$$n = \left\lceil \left\{ P_{set}(t) - NP_i(f = 2.1, u = u_{uni} \right\} \Big/ \left\{ \frac{r(t)P_i(f = 2.0, u = u_{max}(2.0))}{u_{max}(2.0)Ncap_{max}(f = 2.0)} + \left[1 - \frac{r(t)}{u_{max}(2.0)Ncap_{max}(f = 2.0)} \right] P_0 - P_i(f = 2.1, u = u_{uni}) \right\} \right\rceil \tag{3.63}$$

n' can then be derived from Equation (3.62). As shown in the Lemma III of 3.3.6, by increasing n, the total power consumption can be continuously reduced from $P_{uni}(t)$ to the power consumption lower bound in equation (3.95) with an error less than the power consumption of one server. The approximated minimum power consumption is reached when $n = N$, i.e. all workload are packed to servers running at 2.0 GHz.

CASE 3: When $P_{set}(t) < P_{uni}(t)$ and $r(t) > u_{max}(2.0) \times N \times cap_{max}(f = 2.0)$, the data center's operating strategy works as follows. The combination of a low power set point and a large number of requests pushes the utilization rate of the servers to the upper limit.

Note that even by running all servers at 2.0 GHz with the utilization rate at the upper limit which satisfies the SLA, the data center can only handle $N \times u_{max}(2.0) \times$

Algorithm 3 Rule-based data center power consumption control Strategy

1: Receives data center power set point $P_{set}(t) = P_{DC}^{base}(t) + B_{cap}(t)RegD(t)$.

2: Calculate total server power consumption with uniformly distributed requests at the highest CPU frequency as $P_{DC}(u_{uni}(t), f_{max}) = \sum_i^N P_i(f = 2.1, u = u_{uni}(t))$.

3: **if** $P_{set}(t) \geq P_{DC}(u_{uni}(t), f_{max})$ **then**

4: Add dummy load evenly to all servers to increase utilization rate as in (3.57) - (3.60)

5: **else if** $r(t) < Nu_{max}(2.0)cap_{max}(f = 2.0)$ **then**

6: Calculate n', the number of servers operating at frequency 2.0 GHz and utilization rate $u_{max}(2.0)$, and $n - n'$, the number of servers idling, as in (3.62) - (3.63). The remaining servers will be operating at frequency 2.1 GHz and utilization rate of $u = u_{uni}(t)$.

7: **else**

8: Calculate n, the number of servers operating at frequency 2.0 GHz and utilization rate $u_{max}(2.0)$ as in (3.67). The rest of the servers are operating at frequency 2.1 GHz and utilization rate as in (3.65).

9: **end if**

$cap_{max}(f = 2.0)$ requests per second. This is smaller than the number of $r(t)$ in CASE 3. Therefore, only n out of a total of N servers can operate at 2.0 GHz with a utilization rate of $u_{max}(2.0)$. The remaining $n(r(t)/N - u_{max}(2.0)cap_{max}(f = 2.0))$ workload will be evenly distributed to the remaining $N - n$ servers operating at a frequency of 2.1 GHz.

The number of servers, n, operating at 2.0 GHz can be calculated online as follows.

The total energy consumption of the data center in CASE 3 include energy consumption from n servers at frequency 2.0 GHz and the utilization rate of $u_{max}(2.0)$, and $N - n$ servers operating at frequency 2.1 GHz with utilization rate of $u(t)$.

$$P_{DC}(t) = \sum_{i=1}^{n} P_i(f = 2.0, u_{max}(2.0)) + \sum_{i=n+1}^{N} P_i(f = 2.1, u(t)) \tag{3.64}$$

where

$$u(t) = \frac{r(t) - n \times u_{max}(2.0)cap_{max}(f = 2.0)}{(N - n)cap_{max}(f = 2.1)} \tag{3.65}$$

The utilization rate of the $N-n$ servers operating at 2.1 GHz satisfies the following relationship:

$$u(t) > u_{uni}(t) > \frac{u_{max}(2.0)cap_{max}(f = 2.0)}{cap_{max}(f = 2.1)} > 0.1 \tag{3.66}$$

Therefore, by setting $P_{set}(t) = P_{DC}(t)$ and combining Equations (3.64) and (3.65), the closed-form solution for n can be derived as:

$$\begin{aligned} n = \Bigg\lfloor \Big\{ N(2.1\alpha_3^4 + \alpha_4^4) + \frac{(2.1\alpha_1^4 + \alpha_2^4)r}{cap_{max}(f = 2.1)} - P_{set}(t) \Big\} \Big/ \\ \Big\{ 2.1\alpha_3^4 + \alpha_4^4 + (2.1\alpha_1^4 + \alpha_2^4) \frac{u_{max}(2.0)cap_{max}(f = 2.0)}{cap_{max}(f = 2.1)} \\ - P_i(f = 2.0, u = u_{max}(2.0)) \Big\} \Bigg\rfloor \end{aligned} \tag{3.67}$$

As shown in the Lemma IV of 3.3.6, by gradually increasing n, the total power consumption can be reduced from $P_{uni}(t)$ to the power consumption lower bound in equation (3.101) with an error less than the power consumption of one server. The approximated minimum power consumption is reached when the $n = N - n^*_{2.1}$, where $n^*_{2.1}$ is defined in Equation (3.84). In summary, the rule-based data center power consumption control strategy is presented in Algorithm 3.

3.3.4 Numerical Study

A. Simulation Setup

It is assumed that the data center in the numerical study has 100,000 servers. Given estimates of data center size from 2017 [110] ranging from 50,000 - 80,000, we believe that 100,000 servers in a data center is reasonable to simulate a large data center. The servers are assumed to have the same power curves as shown in Fig. 3.6 with power consumption ranging from 22W to 85W. The maximum capacity of each server is 1230 requests per second. The SLA specifies that 90% of the requests will be processed within 115 ms. The corresponding limit on the utilization rate are 0.8 at 2.1 GHz and 0.77 at 2.0 GHz. To simulate the data center's workload, we adopted Wikipedia's access trace from the online repository [111]. The historical prices for frequency regulation and energy from the PJM market are used for electricity market simulation. According to the data center efficiency reports [112, 113], the average power usage effectiveness (PUE) of Google data centers is about 1.12, i.e., the non-IT load is about 12% of IT load. The PUEs of Google and Facebook data centers are as low as 1.07. Assuming that cooling is the major non-IT

load, η_{cool} is set to be 0.12 in the following simulation.

B. Performance of Data Center Requests Forecast

The accuracy of the data center requests forecast is crucial to determining the optimal level of bidding quantity for both energy and frequency regulation service. In the numerical study, the Wikipedia access trace is adopted to simulate the data center's workload. Wikipedia is one of the most visited websites on the Internet, and it is hosted on more than 350 servers [114]. The Wikipedia's access trace contains about 10 percent of all users' requests to Wikipedia, which is collected over about 32 days with a granularity of a millisecond.

Table 3.4: Forecast performance comparison among English, Spanish, and Polish web pages

	English Page	Spanish Page	Polish Page
MAPE	4.09%	7.37%	11.70%
Normalized Standard Deviation	3.58%	10.28%	21.79%

The requests for visiting English, Spanish and Polish web pages are used to evaluate the performance of the data center request forecast. The snapshots of the three groups of web page access data are shown in Fig.3.8.

An ARIMA model is built to perform rolling hour-ahead data center workload trace prediction. The last week of the workload trace of Wikipedia data is used for out-of-sample testing. The forecasted and actual workload trace are depicted in Fig. 3.9. As shown in Fig. 3.9, the hour-ahead workload trace prediction is fairly accurate. To quantify

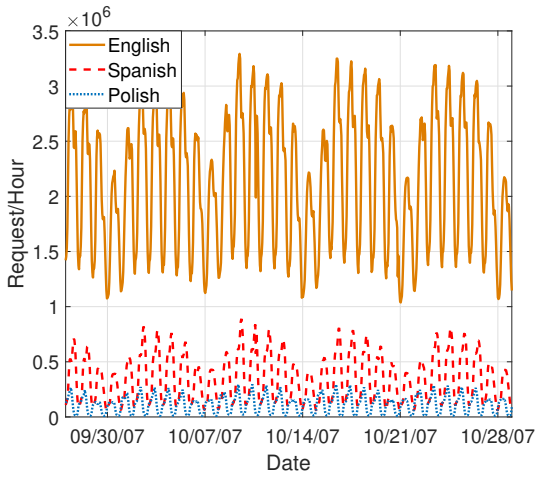

Figure 3.8: Request traces of English, Spanish and Polish pages

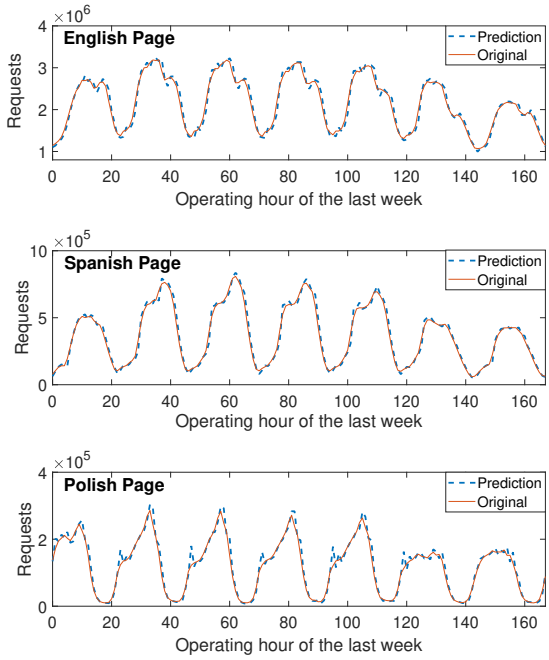

Figure 3.9: Prediction of English, Spanish and Polish page visits

the performance of the forecast method, the mean absolute percentage error (MAPE) of the hour-ahead workload forecast and the standard deviation normalized over mean of the

5-minute workload data of the last week are shown in Table 3.4. As shown in the table, the forecast accuracy for the English web page is the highest with a MAPE smaller than 5%. Because the normalized standard deviation of the number of visits for the Spanish and Polish pages are higher, the corresponding prediction errors are also larger.

In the following simulations, the requests arrival rate is scaled up so that the peak hourly utilization rate for the data center with 100,000 host servers is about 65% when the CPU operating frequency is 2.1 GHz.

C. Performance of the Electricity Price Forecast

The net earnings for a data center to participate in the frequency regulation mar-

Table 3.5: Extracted features

Features	
Last-4-hour Prices	Energy Prices
	Regulation Capacity Clearing Prices
	Performance Clearing Prices
	Mileage Ratio
Last-4-hour Generation	Solar Generation
	Wind Generation
	Storage Generation
	Hydro Generation
	Other Renewable
	Nuclear Generation
	Coal Generation
	Oil Generation
	Gas Generation
	Multiple Fuels
	Other Generation
Last-4-hour Load	Total Demand
Time	Operating Hour of the Day

ket depend on the difference between the frequency regulation service price and the energy

price. When the frequency regulation service price is higher than the energy price, i.e., the price difference is positive, the data center receives extra benefits by providing frequency regulation services compared to simply minimizing its power consumption. Hence, the accuracy of the price difference prediction is crucial to the successful implementation of the frequency regulation provision strategy for data centers.

Table 3.6: Performance of the price difference forecast

	Training	Validation	Testing
F1 score	0.73	0.68	0.70

The historical prices for frequency regulation services and energy from the PJM market is leveraged to build and test the price difference forecasting algorithms. The explanatory variables in the price forecast model include the hourly electricity demand, hourly electricity generation by energy source, and hour of the day. The input features of the price forecast model are summarized in Table 3.5. The data sets from the last week of each month in 2017 are chosen as the test set. The remaining data in 2017 are used for training and validation with a split-ratio of 0.7/0.3. Feed-forward neural network with batch normalization and two hidden layers of 256 neurons and 128 neurons respectively is trained. Early stopping [115] technique is adopted based on the F1 score of the binary classification problem for predicting the sign of the price difference. F1 score is a widely used metric for binary classification. The positive class corresponds to the case where the frequency regulation price is larger than the effective energy price, while the negative class corresponds to the case where the regulation price is smaller than the effective energy price. The performance score of the signal following is assumed to be 100% here. Although the forecast for the

magnitude of price difference is necessary for the risk estimation, the sign of the frequency regulation price minus the effective energy price is more important in terms of the total gain. As reported in Table 3.6, the F1 score is 0.70 in the test set. Note that, for the false negative cases, i.e., the expected price difference is wrongly predicted to be negative, the data center will operate in the power minimization mode, which does not incur extra cost.

D. Performance of Frequency Regulation Service Provision by s Data Center

The performance of the frequency regulation service provision by a data center will be evaluated from three perspectives: frequency regulation signal following performance, electricity cost, and request response time. The price prediction result of the 12 last weeks in each month of the year 2017 is used in the simulation. During the performance evaluation, we assume that the data center will provide frequency regulation service to the electricity market whenever the expected frequency regulation service price is higher than the energy price. In other words, we do not consider the risk constraint. In the real-time operation simulations, the data center is expected to follow the historical frequency regulation signals from the PJM market. The requests served by the data are derived from the scaled requests arrival rate of English, Spanish and Polish pages in the last week of Wikipedia trace as shown in Fig. 3.10, which is repeatedly used. The utilization rate of each server is determined by Algorithm 3 with the bi-linear power model. The actual power consumption of the data center is estimated with empirical measurement data with interpolation.

The frequency regulation signal following performance of the proposed data center power consumption control algorithm is quantified by three metrics: accuracy, delay, and

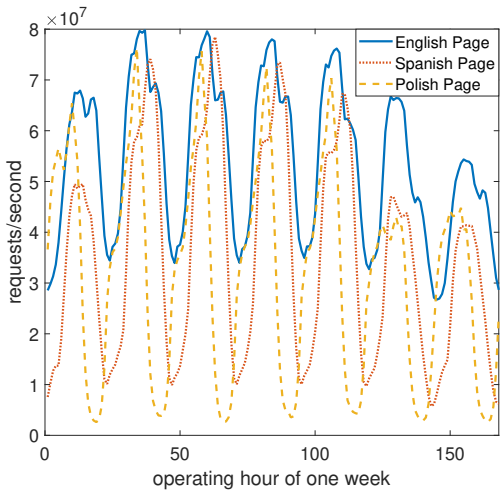

Figure 3.10: Hourly-averaged request arrival rate after scaling

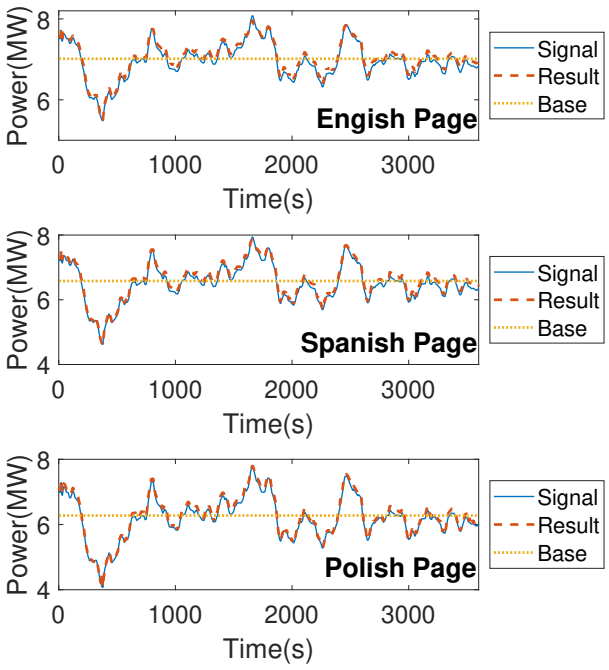

Figure 3.11: Frequency regulation signal following of one hour for the three different pages

precision. The frequency regulation signal and the actual power consumption trajectory of the data center for an hour is depicted in Fig. 3.11. It can be seen from the figure that the proposed data center power consumption control algorithm allows the data center to closely follow the frequency regulation signals. The accuracy and precision scores for the 12 weeks are calculated and shown in Table 3.7. The power set point can be determined by the rule based control algorithm within about 0.2s with a MATLAB program on a standard Dell desktop computer. The delay scores are all 100% for all the test cases. It can be seen that the accuracy and precision scores of the data center are very high for the three types of page visit traces. The small frequency regulation signal tracking errors mainly come from two sources: the requests prediction error and the approximation error of the piece-wise bi-linear server power model. For comparison, the signal following performance of the benchmarks without dummy load and using a linear power model [89] are also provided. When the dummy load control knob is removed, the data center is unable to closely follow the frequency regulation signal when the power set point is higher than the maximum power consumption under uniform request routing. Thus, the introduction of dummy load increases the range of power consumption of the data center. This allows the data center to meet the power set points that frequency scaling alone cannot meet. In terms of using a linear power model, the accuracy score remains reasonable as the correlation is still high. However, the large power estimation error of the linear model results in a significant drop in precision score.

We now present the reduction in electricity cost by participating in the frequency regulation market for the data center. If the data center does not provide frequency regu-

Table 3.7: Frequency regulation signal following performance scores

Dummy Load	Power Model	Performance Score	English Page	Spanish Page	Polish Page
Yes	Bilinear	Accuracy	99.76%	99.69%	99.62%
		Precision	95.35%	95.87%	95.74%
	Linear	Accuracy	96.37%	96.89%	96.28%
		Precision	52.40%	56.99%	55.38%
No	Bilinear	Accuracy	99.58%	98.87%	95.44%
		Precision	92.59%	92.54%	92.36%
	Linear	Accuracy	43.97%	52.45%	42.21%
		Precision	8.08%	20.04%	20.34%

lation service to the power system, then it will operate to minimize its power consumption. The electricity costs of the data center with the proposed algorithm and benchmarks including removing the control knob of the dummy load and the power minimization strategy are reported in Table 3.8. For a data center with 100,000 servers, the proposed data center control algorithm results in a $21,590 (8.1%) electricity costs reduction for the 12 weeks on average compared to the power minimization strategy. With the introduction of dummy load, a higher upper bound of feasible power consumption can be achieved, which improves the cost saving of frequency regulation by about 300% compared to the case without dummy load as shown in Table 3.8. Moreover, the revenue from providing frequency regulation services with the linear power model cannot cover the increased electricity cost due to poor signal following performance. Hence, the introduction of dummy load and the adoption of a bi-linear model are crucial to the profitability of the data center frequency regulation service provision.

The requests response time of the data center when providing frequency regulation

Table 3.8: Energy costs with/without providing frequency regulation service

	Dummy Load	Power Model	English Page	Spanish Page	Polish Page
Cost With Frequency Regulation ($)	Yes	Bilinear	298.38K	238.29K	217.91K
		Linear	326.83K	271.15K	251.08K
	No	Bilinear	309.92K	254.31K	233.49K
		Linear	337.16K	278.01K	252.41K
Costs with Minimum Power($)			316.65K	262.62K	240.08K

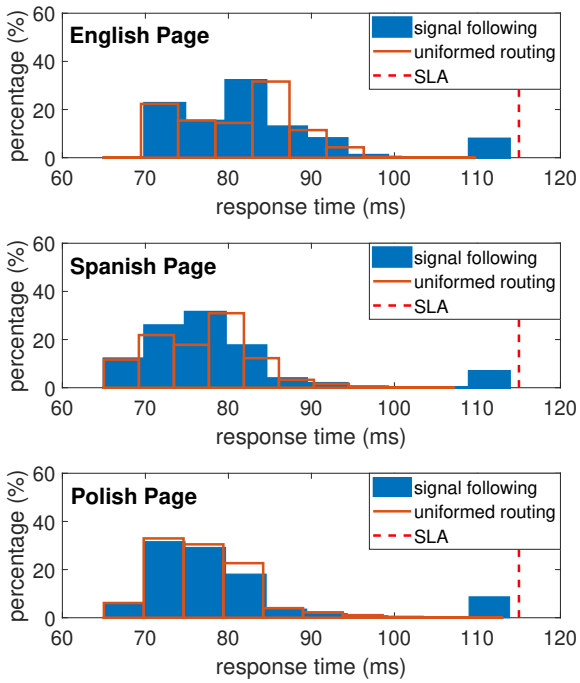

Figure 3.12: Distribution of requests response time

services is calculated based on the proposed request routing algorithm. The distribution of request response time during the hours when the data center provides frequency regulation is shown in Fig. 3.12. Compared to the uniform request routing strategy, when the data center

follows the frequency regulation signals, only a small portion of requests' response time moved closer to the SLA's response time limit. If the data center does not provide frequency regulation service and instead minimizes power consumption with a packing strategy, then the response time of almost all the requests will be very close to the SLA's response time limit. Hence, compared to the minimum power consumption control strategy, the proposed data center control with frequency regulation provision reduces not only electricity costs, but also the response time of requests.

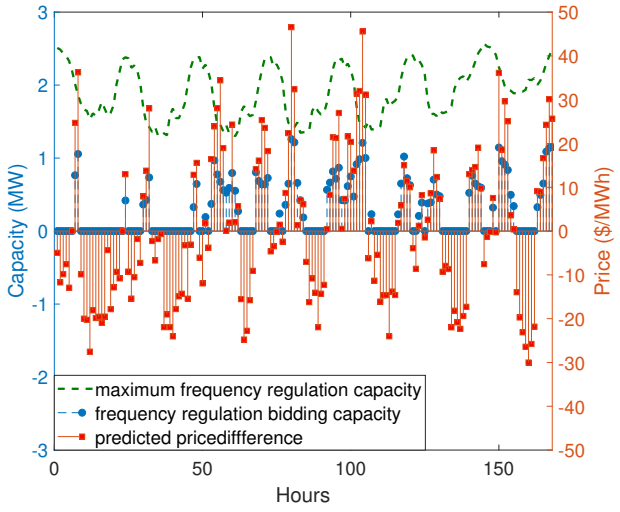

Figure 3.13: Frequency regulation service bidding capacity versus predicted price difference with weekly risk limit of $336

E. Impacts of Risk Limit on Frequency Regulation Bidding Capacity and Net Earnings

The impacts of risk limit on frequency regulation bidding capacity and the data center's net earnings are evaluated in this subsection. As shown in the earlier subsection, the performance scores of the data center in following frequency regulation signals are almost

perfect, hence the performance scores are assumed to be 100% in the evaluation here. For illustrative purpose, only the scaled English page traces are used in the simulation in this subsection. By setting the weekly bidding risk limit δ_{risk} at \$336, the frequency regulation bidding quantities and the differences between frequency regulation prices and energy prices are shown in Fig. 3.13 for one sample week. In the figure, the green dashed line represents the maximum feasible frequency regulation bidding capacity calculated based on the predicted requests arrival rate of the next hour. The red squares denote the predicted hourly price differences, while the blue dots are the frequency regulation bidding capacity obtained from the risk-limited data center bidding strategy. It can be seen from the figure that the bidding capacities are zero when the expected price differences are negative. In addition, the actual bidding capacity is scaled down from the maximum feasible bidding capacity when the confidence level in the positive price difference forecast is low. Next,

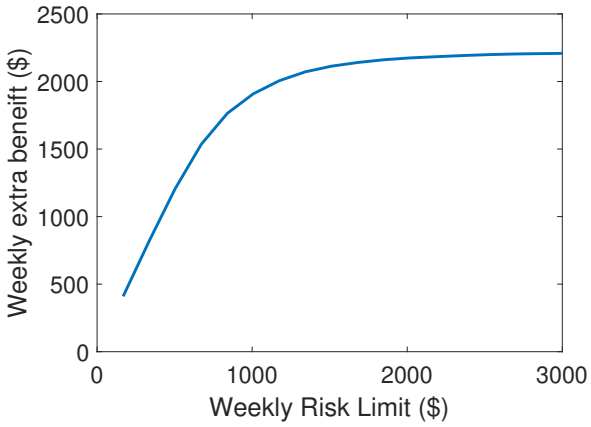

Figure 3.14: Weekly extra net earnings versus the risk limit

we gradually increased the weekly bidding risk limit from \$0 to \$3,000 and recorded the extra earnings of the data center by providing frequency regulation services compared to

the benchmark case where the data center minimizes power consumption. The trade-off between the weekly risk limit and the extra net earnings are depicted in Fig. 3.14. As shown in the figure, the extra net earnings made by the data center increases with the bidding risk limit. However, when the risk limit is very high, the saturation effect kicks in, which leads to a slow increase in extra net earnings with the risk limit.

3.3.5 Conclusion

The operational flexibility of the data centers can be leveraged to provide valuable frequency regulation services in the smart grid. A comprehensive frequency regulation service provision framework is proposed in the section. A risk constrained hour-ahead bidding strategy and a real-time frequency regulation signal following algorithm are developed. The introduction of dummy load and the realistic server power consumption model allow data centers to follow real-world frequency regulation signals with over 95% accuracy. Numerical study with Wikipedia's access trace shows that with reliable energy and frequency regulation service price forecast, data centers can reduce their electricity bill by more than 8% without violating service level agreements.

3.3.6 Proofs

Derivative of Risk Function

Lemma 1 *If the frequency regulation service price is positive, then the derivative of risk function with respect to power consumption base is greater than or equal to zero.*

Proof. For any $\Delta P > 0$,

$$f_{risk}(P_{DC}^{base}(t) + \Delta P) = \iint_{V'} Pr(C_{reg}(t), C_{efe}(t)) \big\{ C_{efe}(t)[P_{DC}^{base}(t) + \Delta P$$

$$- P_{DC}^{min}(t)] - C_{reg}(t) score(t) B_{cap}(t) \big\} dC_{reg}(t) dC_{efe}(t) \tag{3.68}$$

where the integral region is

$$\begin{aligned} V' = & \{C_{reg}(t), C_{efe}(t) | C_{efe}(t)[P_{DC}^{base}(t) + \Delta P - P_{DC}^{min}(t)] \\ & > C_{reg}(t) score(t) B_{cap}(t)\} \end{aligned} \tag{3.69}$$

when $P_{DC}^{base}(t) = P_{DC}^{min}(t)$, we have $B_{cap}(t) = 0$. This is because $B_{cap}(t) \geq 0$ and $P_{DC}^{base}(t) - B_{cap}(t) \geq P_{DC}^{min}(t)$.

In this case,

$$f_{risk}(P_{DC}^{base}(t) + \Delta P) \geq f_{risk}(P_{DC}^{base}(t)) = 0 \tag{3.70}$$

When $P_{DC}^{base}(t) > P_{DC}^{min}(t)$,

$$V = \{C_{reg}(t), C_{efe}(t) | C_{efe}(t) > \frac{C_{reg}(t) score(t) B_{cap}(t)}{P_{DC}^{base}(t) - P_{DC}^{min}(t)}\} \tag{3.71}$$

$$V' = \{C_{reg}(t), C_{efe}(t) | C_{efe}(t) > \frac{C_{reg}(t) score(t) B_{cap}(t)}{P_{DC}^{base}(t) + \Delta P - P_{DC}^{min}(t)}\} \tag{3.72}$$

We can see that, with $C_{reg}(t) > 0$, $V \subseteq V'$. Therefore, we have

$$\begin{aligned} & f_{risk}(P_{DC}^{base}(t) + \Delta P) - f_{risk}(P_{DC}^{base}(t)) \\ & = \iint_{V'-V} Pr(C_{reg}(t), C_{efe}(t)) \big\{ C_{efe}(t)[P_{DC}^{base}(t) + \Delta P \\ & - P_{DC}^{min}(t)] - C_{reg}(t) score(t) B_{cap}(t) \big\} dC_{reg}(t) dC_{efe}(t) \\ & + \iint_{V} Pr(C_{reg}(t), C_{efe}(t)) C_{efe}(t) \Delta P dC_{reg}(t) dC_{efe}(t) \\ & \geq 0 \end{aligned} \tag{3.73}$$

Hence,

$$\frac{\partial f_{risk}}{\partial P_{DC}^{base}(t)} = \lim_{\Delta P \to 0} \frac{f_{risk}(P_{DC}^{base}(t) + \Delta P) - f_{risk}(P_{DC}^{base}(t))}{\Delta P} \geq 0 \tag{3.74}$$

■

Proof of the Minimum Response Time

Lemma 2 *The minimum total request response time is achieved when the workload is uniformly distributed to all servers running at 2.1 GHz.*

Proof. The response time decreases with the increase of frequency for a given utilization rate. The utilization rate also decreases with the increase of frequency for a given amount of requests. Hence, the response time decreases with the increase of frequency at each server.

$$\begin{aligned} &\sum_{i=1}^{N} r_i(t) rt_i(t) = \sum_{i=1}^{N} r_i(t) f_{rt}(f_i(t), \frac{r_i(t)}{cap_{max}(f_i(t))}) \\ &\geq \sum_{i=1}^{N} r_i(t) f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})}) \end{aligned} \tag{3.75}$$

Ignoring the frequency regulation signal following constraint, the total request response time minimization problem can be reformulated as

$$\min \sum_{i=1}^{N} r_i(t) f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})}) \tag{3.76}$$

s.t.

$$\sum_{i=1}^{N} r_i(t) = r(t) \tag{3.77}$$

Define $f_{rt,2.1} = f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})})$. Then it can be seen from Fig. 3.7 that

$$\frac{d\, f_{rt,2.1}}{d\, r_i(t)} = \frac{1}{cap_{max}(f_{max})} \frac{d\, f_{rt,2.1}}{d\, u_i(t)} \geq 0 \tag{3.78}$$

$$\frac{d^2 f_{rt,2.1}}{(d\, r_i(t))^2} = \frac{1}{(cap_{max}(f_{max}))^2} \frac{d^2 f_{rt,2.1}}{(d\, u_i(t))^2} \geq 0 \tag{3.79}$$

Thus,

$$\frac{d^2 r_i(t) f_{rt,2.1}}{(d\, r_i(t))^2} = 2\frac{d\, f_{rt,2.1}}{d\, r_i(t)} + r_i(t)\frac{d^2 f_{rt,2.1}}{(d\, r_i(t))^2} \geq 0 \tag{3.80}$$

Therefore, the objective function (3.76), which is the sum of $r_i(t) f_{rt,2.1}$, is convex.

The Lagrange function of problem (3.76)-(3.77) is:

$$\mathcal{L} = \sum_{i=1}^{N} r_i(t) f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})}) + \lambda(r(t) - \sum_{i=1}^{N} r_i(t)) \tag{3.81}$$

where λ is the Lagrange multiplier.

By taking partial derivative with respect to $r_i(t)$,

$$\frac{\partial \mathcal{L}}{\partial r_i(t)} = f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})}) + r_i(t)\frac{\partial f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})})}{\partial r_i(t)} - \lambda \tag{3.82}$$

At optimal solutions, (3.82) equals zero. Hence, we have

$$f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})}) + r_i(t)\ \frac{\partial f_{rt}(f = 2.1, \frac{r_i(t)}{cap_{max}(f_{max})})}{\partial r_i(t)} = \lambda, \forall i \tag{3.83}$$

The above optimality condition is achieved by uniformed routing, i.e. $r_i(t) = r_j(t), \forall i, j$. ■

Packing Strategy Monotonically Reduces Power Consumption to the Minimum Power Consumption

Lemma 3 *When $r(t) \leq u_{max}(2.0) \times N \times cap_{max}(2.0)$, starting from the baseline operating strategy where the requests are uniformly routed to all servers running at $f = 2.1$ GHz, the*

total power consumption can be reduced by selecting n servers whose workload are packed to n' servers running at 2.0 GHz with the maximum utilization rate $u_{max}(2.0)$ as in equation (3.61). By gradually increasing n, the power consumption can be monotonically reduced to the minimum power consumption with error less than the power consumption of one server, which is achieved when $n = N$.

Lemma 4 *When $r(t) > u_{max}(2.0) \times N \times cap_{max}(2.0)$, starting from the baseline operating strategy where the requests are uniformly routed to all servers running at $f = 2.1$ GHz, the total power consumption can be reduced by curtailing workload on n servers to operate at 2.0 GHz with the maximum utilization rate $u_{max}(2.0)$ and uniformly distributing the remaining workload on the $N - n$ servers as in Equation (3.64). By gradually increasing n, the power consumption can be monotonically reduced to the minimum power consumption with error less than the power consumption of two servers, which is achieved when $n = N - n_{2.1}*$, where*

$$n^*_{2.1} = \frac{r(t) - u_{max}(2.0)Ncap_{max}(2.0)}{u_{max}(2.1)cap_{max}(2.1) - u_{max}(2.0)cap_{max}(2.0)} \tag{3.84}$$

Proof. The total power consumption of a data center is:

$$\begin{aligned} P_{DC}(t) &= \sum_{i=1}^{N} P_i(t) \\ &= \sum_{i=1}^{N}(P_i(t) - P_0) + \sum_{i=1}^{N}(P_0) \end{aligned} \tag{3.85}$$

The dynamic power consumption per request per second for server i is defined as

$$dpr_i(t) = \frac{P_i(t) - P_0}{r_i(t)} \tag{3.86}$$

Substitute the energy consumption model (3.23) into (3.86):

$$\begin{aligned} dpr_i(t) =& \frac{\alpha_1^j f_i(t) u_i(t) + \alpha_2^j u_i(t) + \alpha_3^j f_i(t) + \alpha_4^j - P_0}{r_i(t)} \\ =& \frac{\alpha_1^j f_i(t) u_i(t) + \alpha_2^j u_i(t) + \alpha_3^j f_i(t) + \alpha_4^j - P_0}{cap_{max}(f_i(t)) u_i(t)} \\ =& \frac{\alpha_1^j f_i(t) u_i(t) + \alpha_2^j u_i(t) + \alpha_3^j f_i(t) + \alpha_4^j - P_0}{\frac{cap_{max}}{f_{max}} f_i(t) u_i(t)} \end{aligned} \tag{3.87}$$

Equation (3.86) can be transformed into the following form:

$$dpr_i(t) = \frac{\alpha_1^j}{cap_f} + \frac{\alpha_2^j}{cap_f f_i(t)} + \frac{\alpha_3^j + \frac{\alpha_4^j - P_0}{f_i(t)}}{cap_f u_i(t)} \tag{3.88}$$

where $cap_f = cap_{max}/f_{max}$. By plugging $u_i(t) = 0$ into Equation (3.23), we get $\alpha_3^j f_i(t) + \alpha_4^j - P_0 \geq 0$ or equivalently:

$$\alpha_3^j + \frac{\alpha_4^j - P_0}{f_i(t)} \geq 0 \tag{3.89}$$

Therefore, for a given CPU frequency, the first two terms on the right-hand side of (3.88) are constant, and the last term decreases with the increase of $u_i(t)$. In other words, the dynamic power consumption per request for server i, $dpr_i(t)$, decreases with higher utilization rate.

It can be seen in Fig. 3.15 that for a given utilization rate, the dynamic power consumption per request, dpr_i, roughly stays at the same value when $1.2 \leq f \leq 2.0$. However, as shown in Fig. 3.7, for a given SLA limit rt_{SLA}, a higher utilization rate can be reached when the CPU frequency increases from 1.2 GHz to 2.0 GHz. In other words, $u_{max}(f = 2.0) > u_{max}(f), \forall f \in [1.2, 2.0)$. As the $dpr_i(t)$ decreases with higher $u_i(t)$, $dpr_i(f = 2.0, u = u_{max}(f = 2.0)) < dpr_i(f, u = u_{max}(f)), \forall f \in [1.2, 2.0)$.

It can also be seen in Fig. 3.15 that dpr_i jumps when f increases from 2.0 GHz to 2.1 GHz. By performing a calculation with server power consumption curve and dynamic power curve parameters, we can verify that $dpr_i(f = 2.1,$

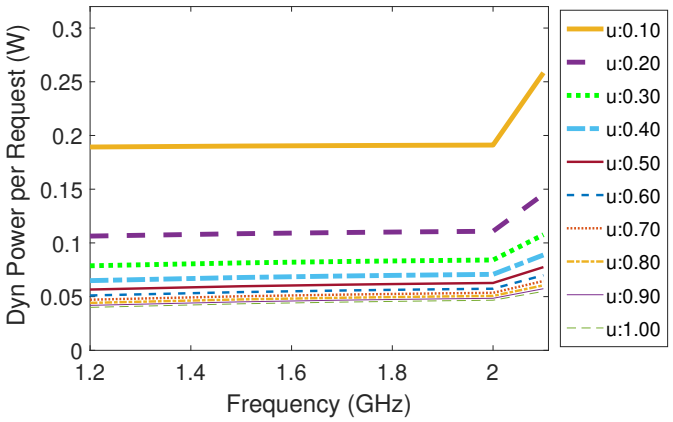

Figure 3.15: Dynamic power consumption per request

$u = u_{max}(f = 2.1)) > dpr_i(f = 2.0, u = u_{max}(f = 2.0))$. Therefore, the minimum dynamic power consumption per request of each server, $dpr_i^{min}(t)$, is achieved when the requests are packed to fully utilize the servers operating with CPU frequency of 2.0 GHz.

For Lemma 3: If the total requests arrival rate falls into the following range, $r(t) \leq u_{max}(2.0) \times N \times cap_{max}(2.0)$, we can pack the workload of n servers to n' servers with $f_i(t) = 2.0$ and $u_i(t) = u_{max}(2.0)$ according to Equation (3.62).

Let's select two different numbers of servers n_1 and n_2 for packing, where $n_1 < n_2$. We will be packing the workload of n_1 (n_2) servers to n_1' (n_2') servers running at 2.0 GHz with a utilization rate of $u_{max}(2.0)$. n_1' and n_2' can be calculated as follows:

$$n_1' = \left\lceil \frac{n_1 r(t)}{u_{max}(2.0) N cap_{max}(f = 2.0)} \right\rceil \tag{3.90}$$

$$n_2' = \left\lceil \frac{n_2 r(t)}{u_{max}(2.0) N cap_{max}(f = 2.0)} \right\rceil \tag{3.91}$$

Thus, $n_1' \leq n_2'$. Let's define the workload handled by he n_1' and n_2' servers as

$$r_1(t) = n_1' u_{max}(2.0) cap_{max}(2.0) \tag{3.92}$$

$$r_2(t) = n_2' u_{max}(2.0) cap_{max}(2.0) \tag{3.93}$$

Since $n_1' \leq n_2'$, we have $r_1(t) \leq r_2(t)$. Because $dpr_i(f = 2.1, u = u_{uni}(t)) > dpr_i^{min}(t)$, we can show that

$$\begin{aligned} P_{uni}(t) &= \sum_{i=1}^{n} (P_i(t) - P_0) + \sum_{i=1}^{n} P_0 \\ &= dpr_i(f = 2.1, u = u_{uni}(t)) r(t) + \sum_{i=1}^{N} P_0 \\ &\geq dpr_i^{min}(t) r_1(t) + dpr_i(f = 2.1, u = u_{uni}(t))[r(t) - r_1(t)] + \sum_{i=1}^{N} P_0 \\ &\geq dpr_i^{min}(t) r_2(t) + dpr_i(f = 2.1, u = u_{uni}(t))[r(t) - r_2(t)] + \sum_{i=1}^{N} P_0 \end{aligned} \tag{3.94}$$

Therefore, by gradually increasing n, the power consumption reduces monotonically. For any feasible request routing strategy, we have

$$\begin{aligned} P_{DC}(t) &= \sum_{i=1}^{N} P_i(t) = \sum_{i=1}^{N} (P_i(t) - P_0) + \sum_{i=1}^{N} P_0 \\ &= \sum_{i=1}^{N} dpr_i(t) r_i(t) + \sum_{i=1}^{N} P_0 \\ &\geq dpr_i^{min}(t) \sum_{i=1}^{n_{2.0}} r_i(t) + \sum_{i=1}^{N} P_0 \\ &= dpr_i^{min}(t) r(t) + \sum_{i=1}^{N} P_0 = \underline{P}_1 \end{aligned} \tag{3.95}$$

where $n_{2.0}$ is the number of servers running at 2.0 GHz with a utilization rate of $u_{max}(2.0)$ after packing all the workload.

$$n_{2.0} = \left\lceil \frac{r(t)}{u_{max}(2.0) N cap_{max}(2.0)} \right\rceil \tag{3.96}$$

For Lemma 4: If the total requests arrival rate falls into the following range, $r(t) > u_{max}(2.0) \times N \times cap_{max}(2.0)$, to satisfy the SLA constraint, there must be some servers running at $f = 2.1$ GHz to handle the additional workload.

In this case, the data center power consumption can be reduced by decreasing the workload on n servers by reducing the CPU frequency from 2.1 GHz to 2.0 GHz with the maximum utilization rate $u_{max}(2.0)$. The remaining workload will be uniformly distributed to the $N - n$ servers according to Equation (3.64).

Let's select two different number of servers n_1 and n_2 for workload and CPU frequency reduction, where $n_1 < n_2$.

Define the workload handled by the n_1 and n_2 servers as r_1 and r_2. The workload can be calculated as

$$\begin{aligned} r_1(t) &= n_1 u_{max}(2.0) cap_{max}(2.0) \\ r_2(t) &= n_2 u_{max}(2.0) cap_{max}(2.0) \end{aligned} \tag{3.97}$$

Because $n_1 < n_2$, we have $r_1(t) < r_2(t)$.

Now, let's define the utilization rate of the remaining $N - n_1$ and $N - n_2$ servers as u_1 and u_2. The utilization rates can be calculated as

$$\begin{aligned} u_1(t) &= \frac{n_1(u_{uni}(t) cap_{max}(2.1) - u_{max}(2.0) cap_{max}(2.0))}{N - n_1} + u_{uni}(t) \\ u_2(t) &= \frac{n_2(u_{uni}(t) cap_{max}(2.1) - u_{max}(2.0) cap_{max}(2.0))}{N - n_2} + u_{uni}(t) \end{aligned} \tag{3.98}$$

Therefore, $u_1(t) < u_2(t)$ and it can be seen that

$$dpr_i(f = 2.1, u = u_1(t)) > dpr_i(f = 2.1, u = u_2(t))$$

Now it can be shown that

$$P_{uni}(t) = dpr_i(f = 2.1, u_{uni}(t))r(t) + \sum_{i=1}^{N} P_0$$
$$\geq dpr_i^{min}(t)r_1(t) + dpr_i(f = 2.1, u_1(t))[r(t) - r_1(t)] + \sum_{i=1}^{N} P_0$$
$$\geq dpr_i^{min}(t)r_2(t) + dpr_i(f + 2.1, u_2(t))(t))[r(t) - r_2(t)] + \sum_{i=1}^{N} P_0 \quad (3.99)$$

Therefore, by gradually increasing n, the data center power consumption will decrease monotonically.

When $n = N - n_{2.1}^*$, the minimum power consumption is achieved, where

$$n_{2.1}^* = \left\lceil \frac{r(t) - u_{max}(2.0)Ncap_{max}(2.0)}{u_{max}(2.1)cap_{max}(2.1) - u_{max}(2.0)cap_{max}(2.0)} \right\rceil \quad (3.100)$$

Note that $n_{2.1}^*$ is the minimum number of servers that have to operate at 2.1 GHz.

Denote the minimum dynamic power consumption per request $dpr_i(t)$ at $f = 2.1$ as $dpr_i^{min}(t, f = 2.1)$.

Then, we have

$$P_{DC}(t) \geq dpr_i^{min}(t) \sum_{i=n_{2.1}^*+1}^{N} r_i(t) + \sum_{i=1}^{N} P_0 + dpr_i^{min}(t, f = 2.1) \sum_{i=1}^{n_{2.1}^*} r_i(t) = \underline{P}_2 \quad (3.101)$$

Next we will prove that Equation (3.101) holds and $\underline{P}_2$ is the lower bound of power consumption when $r(t) \geq u_{max}(2.0) \times N \times cap_{max}(2.0)$.

Let's denote the workload handled by the servers running at $f = 2.0$GHz as $r'(t)$. $r'(t)$ can then be calculated as

$$r'(t) = \sum_{i=n_{2.1}^*+1}^{N} r_i(t) = (N - n_{2.1}^*)u_{max}(2.0)cap_{max}(2.0) \quad (3.102)$$

Let's denote the remaining workload to be handled by servers running at $f = 2.1$ as $r^*(t)$. Thus we have

$$r^*(t) = r(t) - r'(t) \tag{3.103}$$

Denote the average dynamic power consumption per request of $r^*(t)$ as $dpr(r^*(t))$ and the average dynamic power consumption per request of $r'(t)$ as $dpr(r'(t))$. As $dpr(r^*(t)) \geq dpr_i^{min}(t, f = 2.1)$ and $dpr(r'(t)) \geq dpr_i^{min}(t)$, then for any feasible operating point, we have

$$\begin{aligned} P_{DC}(t) &= r^*(t)dpr(r^*(t)) + r'(t)dpr(r'(t)) + \sum_{i=1}^{N} P_0 \\ &\geq dpr_i^{min}(t, f = 2.1)r^*(t) + dpr_i^{min}(t)r'(t) + \sum_{i=1}^{N} P_0 \end{aligned} \tag{3.104}$$

■

Proof of the Minimum Amount of Requests with Increased Response Time

Lemma 5 *The amount of requests with increased response time compared to the uniform routing is minimized with the packing strategy in the CASE 2 ($P_{set}(t) < P_{uni}(t)$ and $r(t) \leq u_{max}(2.0) \times N \times cap_{max}(f = 2.0)$) of Section 5.2.*

Proof. For the real-time control strategy of CASE 2 in section 5.2, the response time of the requests on the $N - n$ servers running at $f = 2.1$ GHz is the same as that of the uniformed routing. In order to reduce data center power consumption from the power consumption under uniform routing policy $P_{uni}(t)$ by $\Delta P(t)$, we have to increase the response time for $\Delta r(t)$ requests per second by reducing their dynamic power consumption from $dpr_{uni}(t)$ to $dpr_i(t)'$.

Note that $\Delta r(t)$, $\Delta P(t)$, $dpr_i(t)$, and $dpr_i(t)'$ satisfy the following relationship:

$$\Delta r(t) = \frac{\Delta P(t)}{dpr_{uni}(t) - dpr_i(t)'} \tag{3.105}$$

In order to minimize the number of requests with increased response time per second, $\Delta r(t)$, we have to set $dpr_i(t)'$ at its minimum which is $dpr_i^{min}(t)$.

Now, as shown in the 3.3.6, the minimum dynamic power consumption per request per second can be achieved when the servers are operating at $f = 2.0$ GHz and $u_{max}(2.0)$. This is the same packing strategy for Case 2 in Section 5.2. Therefore, the packing strategy proposed for Case 2 minimizes the number of requests with increased response time. ■

Chapter 4

Data-driven Volt-VAR Control in Power Distribution Systems

Although successful field demonstrations of Volt-VAR Control (VVC) have been reported by many electric utilities, there are still many barriers to the widespread adoption of the technology. One of the most significant barriers is the lack of robust distribution network topology and parameter information, which are required in optimization based VVC approaches. In particular, inaccurate distribution secondary systems' information [19, 20, 21] makes it difficult for VVC to ensure that customers' voltages will stay within the acceptable range. Moreover, the model-based control approaches are not always scalable and may not be applicable in real-time control environments [116, 117]. To overcome the drawbacks of model-based approaches, we propose to solve the VVC problem with data-driven approaches.

In this chapter, we investigate deep reinforcement learning (DRL) algorithms for

VVC, including both the on-policy [28] and off-policy [29] algorithms.

4.1 Related Works

The majority of the existing work on VVC adopt a physical model-based optimization/control approach. Due to space limitation, we focus on summarizing recent advancements of VVC technology, which can be separated into three groups. The first group of literature formulates VVC as deterministic optimization problems. The VVC problem is extended to consider voltage-dependent loads [118] and formulated as a mixed-integer quadratically constrained programming problem. A power electronic device, called soft open point [119], is introduced to achieve real-time VVC together with conventional voltage regulation devices. The coordinated control problem is formulated as a mixed-integer second-order conic programming problem. The VVC problem is formulated as a non-cooperative mixed strategy game [120], which considers flexible loads, electric vehicles, and renewable energy sources. The limit on the number of switching operations of voltage regulating devices is considered in the VVC [121], which is formulated as a mixed-integer nonlinear programming problem.

The second group of literature explicitly incorporates the uncertainties of DERs in the VVC problem formulation. A dual time-scale coordination scheme for slow and fast controlling devices is proposed for the VVC problem, which is solved with stochastic [122] and robust [123] optimization algorithms. The model predictive control (MPC) based VVC algorithms are proposed to reduce network losses [124], voltage deviations and excessive wear and tear of voltage regulating devices [125].

To address the algorithm scalability problem and the communication delay issue of the centralized optimization and control approach for VVC, the third group of literature develops non-centralized control schemes, which can be further divided into three subgroups, local VVC algorithms, distributed VVC algorithms, and decentralized VCC algorithms [126]. The local VVC algorithms use only locally available information such as bus voltages to design control strategies. Fully decentralized disturbance-feedback controller [127], gradient-projection algorithm [128], and asynchronous gradient-project algorithm are developed [129] for local voltage controls. The distributed VVC algorithms allow neighboring agents to communicate and share information to cooperatively reach global objectives of VVC. Distributed algorithms such as the alternating direction method of multipliers [130], the dual decomposition method [131], the integral-control-like update scheme [132], and the local optimization and consensus approach [133] are developed to solve VVC problems. The decentralized algorithms are developed with ϵ-decomposition in [134, 135], where centralized control is only needed within the isolated sub-areas.

To remove the dependency on complete and accurate distribution network topology and parameter information, a few researchers have developed reinforcement learning (RL) based algorithms for VVC. The tabular Q-learning algorithm [136] is used to learn the setting of control variables which satisfy operation constraints in power systems. The tabular Q-learning method with the global reward recovered from the consensus-based algorithm [137] is proposed to solve the optimal reactive power dispatch problem. Radial basis functions are used to approximate Q-function in [138] to find the optimal tap settings of the voltage regulation devices. In the existing RL-based algorithms, the VVC problems are

always modeled as Markov decision process (MDP) and solved with Q-learning algorithm, which is a commonly used action-value method in RL. The action-value methods learn to approximate the action-value functions and then select actions based on the estimated action-value functions and the ϵ-greedy algorithm [139].

In Section 4.2, we adopt on-policy policy gradient methods [140, 27] to solve the VVC problem. In Section 4.3, to further improve the sample efficiency, we propose a safe off-policy policy gradient algorithm. Unlike the existing RL-based VVC algorithms, we formulate the VVC problems as a constrained MDP (CMDP) and use policy gradient methods. Policy gradient methods directly learn a parameterized control policy that can select actions without using a value function. Policy gradient methods have two advantages over action-value methods. First, the VVC policy may be a simpler function to approximate than the action-value function. Second, continuous policy parameterization yields stronger convergence guarantees for policy-gradient methods than the ϵ-greedy action selection for action-value methods [139]. Compared to the optimization-based approaches, our proposed algorithm has better scalability and does not require accurate and complete physical model of the distribution network.

4.2 Volt-VAR Control with On-policy Deep Reinforcement Learning Algorithm

4.2.1 Problem Formulation

In this section, we first formulate the VVC problem as an optimization problem and then as a CMDP problem.

A. Volt-VAR Control Formulated as an Optimization Problem

VVC algorithm aims at minimizing the total system losses and equipment operation costs while satisfying voltage constraints. In this formulation, we assume the voltage regulators, on-load tap changers and capacitor banks are the primary control knobs. Then, the VVC problem can be formulated as an optimization problem as follows [141]:

$$\begin{aligned}\min C_p[\sum_{t=1}^{T} P_{loss}^t] + C_r \sum_{t=1}^{T}\sum_{j=1}^{N_r} |Tap_j^r(t) - Tap_j^r(t-1)| \\ + C_l \sum_{t=1}^{T}\sum_{j=1}^{N_l} |Tap_j^l(t) - Tap_j^l(t-1)| \\ + C_c \sum_{t=1}^{T}\sum_{j=1}^{N_c} |Tap_j^c(t) - Tap_j^c(t-1)| \end{aligned} \tag{4.1}$$

s.t.

$$f_{PB}(\boldsymbol{PG}^t, \boldsymbol{QG}^t, \boldsymbol{PD}^t, \boldsymbol{QD}^t, \boldsymbol{TAP}_t, \boldsymbol{u}^t, \boldsymbol{l}^t) = 0, \forall t \tag{4.2}$$

$$f_{OL}(\boldsymbol{PF}^t, \boldsymbol{QF}^t, \boldsymbol{TAP}_t, \boldsymbol{u}^t, \boldsymbol{l}^t) = 0, \forall t \tag{4.3}$$

$$PF_{ij}^{t\,2} + QF_{ij}^{t\,2} = l_{ij}^t u_i^t, \forall i, j \in \mathcal{N}, (i,j) \in E, t \tag{4.4}$$

$$\underline{u} \leq u_i^t \leq \overline{u}, \forall i \in \mathcal{N}, t \tag{4.5}$$

The objective function (4.1) minimizes the total operation costs, which include the costs associated with line losses and the switching costs of voltage regulators, on-load tap changers, and capacitor banks. The switching cost is assumed to be proportional to the absolute number of tap changes between consecutive hours. P_{loss}^t denotes the total real line losses at hour t. C_p, C_r, C_l, and C_c are the cost coefficients for the real power loss, the tap changes of voltage regulators, on-load tap changers, and capacitor banks respectively. N_r, N_l, and N_c are the total numbers of voltage regulators, on-load tap changers, and capacitor

banks. $Tap_j^r(t)$, $Tap_j^l(t)$, and $Tap_j^c(t)$ denote the tap position of the j-th voltage regulator, on-load tap changer, and capacitor bank at hour t. T is the operation horizon of the VVC algorithm.

The formulation of constraints leverages the DistFlow equations [142]. The decision variables of the DistFlow formulation are the vector ($\boldsymbol{u}^t$) of u_i^t for all the nodes ($\mathcal{N}$), the vector ($\boldsymbol{l}^t$) of l_{ij}^t for all the lines (E), and the vector ($\boldsymbol{TAP}_t$) of tap positions for all the devices. u_i^t denotes the square of voltage magnitude of node i at hour t. l_{ij}^t denotes the square of current magnitude of the line connecting node i and j at hour t.

The set of power balance constraints in the DistFlow is represented by (4.2), where $\boldsymbol{PG}^t$, $\boldsymbol{QG}^t$, $\boldsymbol{PD}^t$, and $\boldsymbol{QD}^t$ denote the vector of nodal real and reactive power generations and demands at hour t. The constraints corresponding to the Ohm's law is represented by (4.3), where $\boldsymbol{PF}^t$ and $\boldsymbol{QF}^t$ denote the vector of real and reactive power flows at hour t. Equality constraint (4.4) is the only nonlinear constraint in the DistFlow formulation, which can be relaxed as a second order cone [142]. PF_{ij}^t and QF_{ij}^t are the real and reactive power flow on the line connecting node i and j at hour t. E and $\mathcal{N}$ denote the set of edges and nodes in the distribution feeder. Equation (4.5) represents the nodal voltage constraints, where $\underline{u}$ and $\overline{u}$ are the lower and upper limits for the square of voltage magnitude. The detailed formulations for the operating constraints can be found in [141], where binary variables are introduced to represent the tap positions. The optimization problem shown above is a MICP problem.

Finally, to account for generation and load uncertainties, the VVC problem can be formulated as a MPC [124]. The optimization problem shown above can be solved on a

rolling basis based on the updated load and generation forecasts.

B. Volt-VAR Control Formulated as a Constrained Markov Decision Process

In the Markov decision process (MDP), the grid operator or controller is denoted by an agent. This agent and the distribution grid interact at each of a sequence of discrete time steps $t = 0, 1, 2, \ldots$. At each time step t, the agent receives the system's state $s_t \in \mathcal{S}$, and selects a control action $a_t \in \mathcal{A}(s)$. One time step later, the agent receives a numerical reward $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$, and finds itself in a new state s_{t+1}. The probability of receiving a reward and observing a new state depends on the preceding state and control action as $P(s_{t+1}|s_t, a_t) = P(s_{t+1}|s_0, a_0, ..., s_t, a_t)$.

In the context of the VVC, the state is defined as $s = [\boldsymbol{P}, \boldsymbol{Q}, \mathcal{T}, t]$, where $\boldsymbol{P}, \boldsymbol{Q}, \mathcal{T}$ and t denote the nodal real and reactive power injections, the current tap positions, and the time step. The action taken by a VVC agent is changing the tap positions of controllable devices to $\mathcal{T}'$. The size of the action space is $\Pi_{i=1}^{N_s} n_i$, where $N_s = N_r + N_l + N_c$ is the number of controllable devices and n_i denotes the number of tap positions of device i. The reward received by the controller $R(s_t, a_t, s_{t+1})$ for taking action a_t at state s_t and reaching state s_{t+1} is defined as the negative of the system operational costs, which include the costs associated with real power losses and equipment operations.

$$R(s_t, a_t, s_{t+1}) = -\Big[C_p P_{loss}^t + C_r \sum_{j=1}^{N_r} |Tap_j^r(t+1) - Tap_j^r(t)| + C_l \sum_{j=1}^{N_l} |Tap_j^l(t+1) - Tap_j^l(t)| + C_c \sum_{j=1}^{N_c} |Tap_j^c(t+1) - Tap_j^c(t)|\Big] \tag{4.6}$$

The goal of an agent is to find a control policy π that maximizes the expected

discounted return defined as:

$$J(\pi) = \underset{\tau \sim \pi}{E}[G(\tau)] \tag{4.7}$$

where control policy π is a mapping from state space $\mathcal{S}$ to action space $\mathcal{A}$ for a deterministic policy and a mapping from states to probabilities of selecting each possible action for a probabilistic policy. τ is a trajectory or sequence of states and actions, $\{s_0, a_0, s_1, a_1, ..., s_{T-1}, a_{T-1}, s_T\}$. $G(\tau)$ is the discounted return along a trajectory. $G(\tau) = \sum_{t=0}^{T} \gamma^t R(s_t, a_t, s_{t+1})$, where $\gamma \in (0, 1)$ is the discount factor.

Two important functions, action-value function and state-value function for policy π are defined as follows [139]:

$$Q^{\pi}(s, a) = \underset{\tau \sim \pi}{E}[G(\tau)|s_0 = s, a_0 = a] \tag{4.8}$$

$$V^{\pi}(s) = \underset{\tau \sim \pi}{E}[G(\tau)|s_0 = s] \tag{4.9}$$

The action-value function $Q^{\pi}(s, a)$ represents the expected return starting with state s, taking action a, and following π thereafter. The state-value function $V^{\pi}(s)$ represents the expected return starting with state s and thereafter following policy π.

To enforce the voltage constraints, we augment the MDP with a set of cost functions $R_C(s_t, a_t, s_{t+1})$. For the VVC problem, it is defined as the number of voltage violations across all nodes, i.e.,

$$R_C(s_t, a_t, s_{t+1}) = \sum_{i=1}^{N} [\mathbb{1}(|v_i^{t+1}| > \overline{v}) + \mathbb{1}(|v_i^{t+1}| < \underline{v})] \tag{4.10}$$

where $\mathbb{1}(\cdot)$ is the indicator function; v_i^{t+1} is the voltage of node i at hour $t+1$; $\overline{v}$ and $\underline{v}$ are the upper and lower limits for voltage magnitudes. Additional operating constraints such as the line flow limits could be incorporated in a similar manner.

Now the expected discounted return of policy π with respect to the cost function can be defined as

$$J_C(\pi) = \underset{\tau\sim\pi}{E}[\sum_{t=0}^{T}\gamma^t R_C(s_t, a_t, s_{t+1})] \tag{4.11}$$

The final CMDP formulation for the VVC problem is:

$$\max_{\pi} J(\pi) \tag{4.12}$$

s.t.

$$J_C(\pi) \leq \overline{J} \tag{4.13}$$

where $\overline{J}$ is the limit for the expected discounted return of the cost function associated with the voltage constraints.

4.2.2 Technical Methods

So far all reinforcement learning algorithms adopted to solve the VVC problem have been action-value methods, which approximate the action-value functions through learning and then select actions based on the estimated action-value functions. In this section, we consider policy gradient methods, which learn a parameterized control policy that directly selects actions without consulting a value function [139]. Typically, an approximate policy is parameterized according to the soft-max in action preferences, which makes approaching deterministic policy easier and finding stochastic policy feasible [139]. Both of these goals can not be achieved by the ϵ-greedy action selection in the action-value methods. Another notable advantage of the policy gradient methods over the action-value methods is that the control policy functions may be easier to approximate than action-value functions in many applications such as the VVC problem.

In this section, we first introduce the preliminaries of the policy gradient methods. Then two state-of-the-art policy gradient methods based on trust region algorithms [140, 27] are adopted to solve the VVC problem. Finally, the design of neural networks to approximate the policy and value functions in the two algorithms will be discussed.

A. Preliminaries of policy gradient method

Policy gradient methods learn a parameterized control policy π_θ that maximizes the performance measure $\hat{J}(\pi_\theta)$ by updating the parameter θ iteratively as follows:

$$\theta_{k+1} = \theta_k + \alpha \nabla_\theta \hat{J}(\theta_k) \tag{4.14}$$

According to the policy gradient theorem [139], the gradient can be derived as

$$\nabla_\theta \hat{J}(\theta) = \underset{\tau \sim \pi_\theta}{E}[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)\Psi_t] \tag{4.15}$$

where Ψ may have various forms including the action-value function $Q^{\pi_\theta}(s,a)$ and the advantage function $A^{\pi_\theta}(s,a)$.

The advantage function, which quantifies the improvement by taking action a in state s compared to randomly selecting an action according to policy π_θ and following π_θ afterwards, is defined as

$$A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s) \tag{4.16}$$

Two policy gradient methods, trust region policy optimization (TRPO) and constrained policy optimization (CPO), that use the advantage function are presented in the following subsections. We will discuss how to adopt them to solve the VVC problem formulated as MDP and CMDP. The implementation details of these two algorithms can be found in [140, 27].

B. Trust Region Policy Optimization

Algorithm 4 TRPO for VVC

1: Initialize parameters for policy and value function, θ_0, ϕ_0

2: **for** k = 0,1,2,... **do**

3: Generate sample trajectories $Tr_k = \{\tau\}$ with π_{θ_k} through power flow simulations

4: Calculate the discounted return for the objective $\hat{G}_t$ after each time step t along the trajectories

5: Estimate the advantage for the objective $\hat{A}_t$ based on the value function V_{ϕ_k}

6: Obtain $\pi_{\theta^*_{k+1}}$ by solving (4.18) and (4.19)

7: Update the parameters ϕ_k of the value function neural network with $\hat{G}_t$ as labels

8: **end for**

The TRPO algorithm originally proposed in [140] provides a theoretical guarantee of monotonic improvement of the control policy at each policy iteration step.

The design of the policy iteration procedure is based on the lower bound [27] of the performance improvement of policy $\pi_{\theta'}$ over policy π_θ:

$$J(\pi_{\theta'}) - J(\pi_\theta) \geq \frac{1}{1-\gamma} \underset{\substack{s\sim\eta^{\pi_\theta} \\ a\sim\pi_{\theta'}}}{E} \Big[A^{\pi_\theta}(s,a) - \frac{\gamma \xi^{\pi_{\theta'}}}{1-\gamma} \sqrt{2KL(\pi_{\theta'}||\pi_\theta)[s]} \Big] \tag{4.17}$$

where $\xi^{\pi_{\theta'}} = \max_s |E_{a\sim\pi_{\theta'}}[A^{\pi_\theta}(s,a)]|$. $KL(\pi_{\theta'}||\pi_\theta)[s]$ is the KL-divergence between policy $\pi_{\theta'}$ and π_θ at state s. η^{π_θ} is the discounted future state distribution, $\eta^{\pi_\theta}(s) = (1-\gamma)\sum_{t=0}^{T} \gamma^t P(s_t = s|\pi_\theta)$. $P(s_t = s|\pi_\theta)$ denotes the probability of state s appearing at time t under policy π_θ.

Thus, we can update the policy parameters iteratively by maximizing the expected

advantage with a small step size δ:

$$\pi_{\theta_{k+1}} = arg\max_{\pi_\theta} \underset{\substack{s\sim\eta^{\pi_{\theta_k}} \\ a\sim\pi_\theta}}{E}[A^{\pi_{\theta_k}}(s,a)] \tag{4.18}$$

$$s.t. \underset{s\sim\eta^{\pi_{\theta_k}}}{E}\big[KL(\pi_\theta, \pi_{\theta_k})[s]\big] \leq \delta \tag{4.19}$$

If π_{θ_k} is a feasible solution, the maximum expected advantage is non-negative. With a small enough δ, monotonic policy improvement is guaranteed according to (4.17). The optimization problem (4.18) and (4.19) can be solved by linearizing the objective function and quadratically approximating the KL-divergence around θ_k.

The final iterative TRPO algorithm to solve the VVC problem is shown in Algorithm 4. To adopt the TRPO algorithm for the VVC problem, the reward function is augmented with a penalty term associated with the voltage violations:

$$R'(s_t, a_t, s_{t+1}) = R(s_t, a_t, s_{t+1}) - C_V R_C(s_t, a_t, s_{t+1}) \tag{4.20}$$

where C_V is the penalty factor for voltage violations.

C. Constrained Policy Optimization

To directly solve the VVC problem formulated as a CMDP, the CPO algorithm, which guarantees approximate constraints satisfaction, can be leveraged [27]. The theoretical guarantee of the constraint satisfaction can be shown with the upper bound [27] of the performance improvement associated with constraints of policy $\pi_{\theta'}$ compared to policy π_θ:

$$J_C(\pi_{\theta'}) - J_C(\pi_\theta) \leq \frac{1}{1-\gamma} \underset{\substack{s\sim\eta^{\pi_\theta} \\ a\sim\pi_{\theta'}}}{E}\Big[A_C^{\pi_\theta}(s,a) + \frac{\gamma\xi_C^{\pi_{\theta'}}}{1-\gamma}\sqrt{2KL(\pi_{\theta'}||\pi_\theta)[s]}\Big] \tag{4.21}$$

where $\xi_C^{\pi_{\theta'}} = \max_s |E_{a\sim\pi_{\theta'}}[A_C^{\pi_\theta}(s,a)]|$ and $A_C^{\pi_\theta}(s,a)$ is the corresponding advantage function for the constraint. According to (4.21), the constraint at each updating step is specified as:

$$J_C(\pi_{\theta_k}) + \frac{1}{1-\gamma} \underset{\substack{s\sim\eta^{\pi_{\theta_k}} \\ a\sim\pi_\theta}}{E} [A_C^{\pi_{\theta_k}}(s,a)] \leq \overline{J} \tag{4.22}$$

Algorithm 5 CPO for VVC

1: Initialize parameters for policy and value functions, θ_0, ϕ_0^1, and ϕ_0^2

2: **for** k = 0,1,2,... **do**

3: Generate sample trajectories $Tr_k = \{\tau\}$ with π_{θ_k} through power flow simulations

4: Calculate the discounted returns $\hat{G}_t^1$, $\hat{G}_t^2$ for the objective function and the constraint after each time step t along the trajectories

5: Estimate the advantages for the objective $\hat{A}_t^1$ and the constraint $\hat{A}_t^2$, based on the value functions $V_{\phi_k^1}$ and $V_{\phi_k^2}$

6: **if** the problem (4.18), (4.19) and (4.22) is feasible **then**

7: Obtain the optimal solution $\pi_{\theta_{k+1}^*}$

8: **else**

9: Obtain the solution $\pi_{\theta_{k+1}^*}$ by solving (4.19) and (4.24)

10: **end if**

11: Update the parameters ϕ_k^1 and ϕ_k^2 of the value function neural networks with $\hat{G}_t^1$ and $\hat{G}_t^2$ as labels

12: **end for**

The policy update for CMDP can be found by solving (4.18), (4.19), and (4.22). Therefore, with a small enough δ, the constraint satisfaction is almost guaranteed at step $k+1$ if we start from a feasible solution π_{θ_k} according to (4.21). The worst-case constraint

violation at step $k+1$ is:

$$J_C(\pi_{\theta_{k+1}}) - \overline{J} \leq \frac{\sqrt{2\delta}\gamma\xi^{\pi_{\theta_{k+1}}}}{(1-\gamma)^2} \tag{4.23}$$

Similarly, to solve the optimization problem, (4.22) should be linearized around θ_k. At the beginning of the training process, a feasible solution can be recovered by solving the following problem subject to (4.19):

$$\min_{\pi_\theta} \underset{\substack{s\sim\eta^{\pi_{\theta_k}} \\ a\sim\pi_\theta}}{E}[A_C^{\pi_{\theta_k}}(s,a)] \tag{4.24}$$

The final CPO algorithm to solve the VVC problem is shown in Algorithm 5.

4.2.3 Value and Policy Networks

Both the objective function (4.18) and the expectation of the advantage function associated with the constraint in (4.22) can be calculated with only the state-value function and the policy function as follows:

$$\begin{aligned} &\underset{\substack{s\sim\eta^{\pi_{\theta_k}} \\ a\sim\pi_\theta}}{E}[A^{\pi_{\theta_k}}(s,a)] = \underset{\substack{s\sim\eta^{\pi_{\theta_k}} \\ a\sim\pi_{\theta_k}}}{E}[\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}A^{\pi_{\theta_k}}(s,a)] = \\ &\underset{\substack{s\sim\eta^{\pi_{\theta_k}} \\ a\sim\pi_{\theta_k}}}{E}[\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}(R(s,a,s') + \gamma V^{\pi_{\theta_k}}(s') - V^{\pi_{\theta_k}}(s))] \end{aligned} \tag{4.25}$$

Therefore, we only need to design neural networks to approximate the state-value function and the policy function. The state-value function V_ϕ corresponding to the augmented reward in Algorithm 4 is parameterized with ϕ. The state-value functions corresponding to the reward V_{ϕ_1} and the constraint V_{ϕ_2} in Algorithm 5 are parameterized with ϕ_1 and ϕ_2. The inputs of all the value networks are states. The output is the expected discounted return. The policy function π_θ is approximated by a neural network with pa-

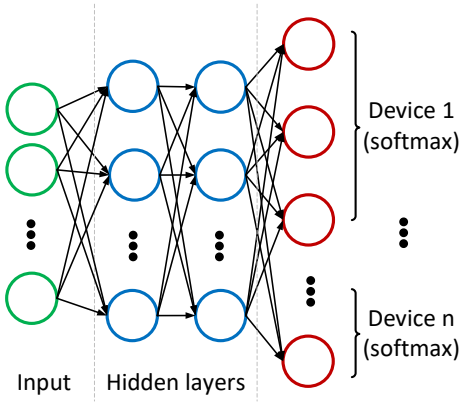

Figure 4.1: Structure of the policy network

rameter θ. The structure of the policy network is shown as in Fig. 4.1. The inputs are the states and the outputs are the probabilities of selecting various actions, which represent the switch status of the devices. The size of the output layer is $\sum_{i=1}^{N_s} n_i$, where N_s and n_i are the number of devices and the number of tap positions for device i. The probability distribution P_i of the actions for device i, is obtained from the subset of the output neurons with size n_i. A softmax activation function is applied to each subset of the output neurons corresponding to a device. The final probability distribution of the tap combinations across all devices is calculated with $P = \Pi_{i=1}^{N_s} P_i$. Thus, in our proposed methods the network size only increases linearly with N_s.

4.2.4 Numerical Study

A. Simulation Setup

The numerical studies are conducted on the IEEE 4-bus and 13-bus distribution test feeders [75]. The real-world smart meter data of an electric utility is used as the nodal

load in the simulation environment to generate power flow solutions. The length of historical data is about six months. One week of data during the summer peak are used for the out-of-sample test and the rest are used for training. The length of the VVC optimization horizon or an episode in reinforcement learning is one week. The load time series data is scaled and allocated to each node according to the load profile of the standard test case. Each test feeder has three switching devices: a voltage regulator, an on-load tap changer, and a capacitor bank. Both the voltage regulator and the on-load tap changer have 11 tap positions with turns ratios between 0.95 and 1.05. The capacitor bank can be switched on and off remotely and the number of 'tap positions' is treated to be 2. The size of the action space for each test case is $11 \times 11 \times 2 = 242$. In the 4-bus test feeder, the capacitor bank is placed at node 4. In the 13-bus test feeder, the capacitor bank is placed at node 675. The nominal capacity of the capacitor banks is $200kW$. Initially, the turns ratios of the voltage regulators and on-load tap changers are 1, while the capacitor banks are switched off. The electricity price C_p is assumed to be $\$40/MWh$. The switching costs of the devices C_r, C_l, and C_c are set at \$0.1 per tap change.

B. Benchmarking Algorithms

The MPC-based optimization algorithm is chosen as the first benchmark. The control horizon is at 24 hours. The ARIMA [143] model is used to forecast the load during the control horizon. The MICP problem formulated in Section 4.2.1 is solved on a rolling basis at each step of MPC. MOSEK and GUROBI are used to solve the MICP problem. The second benchmark is set up by replacing the load forecast with actual load data in the MPC framework. The last benchmark represents the baseline where all switching devices

are kept at their initial positions.

C. Policy Gradient Methods

In the TRPO and CPO algorithms, both the value and policy neural networks have two hidden layers with 64 and 32 neurons respectively. The tanh activation function is used in all the hidden layers. The linear and softmax activation functions are used for the output layers of the state-value and the policy networks. In the TRPO algorithm, the reward function is augmented by a penalty cost for voltage constraint violations. The penalty coefficient C_V is \$1 per voltage violation per node. The terminal state is chosen as the last hour of a week for both algorithms.

D. Performance Comparison

The control performances of CPO, TRPO, and MPC-based approaches are evaluated in this subsection. Both the CPO algorithm and the TRPO algorithm are trained for 500 iterations. Each training iteration consists of 298 episodic trajectories, which correspond to about 50,000 samples. Over the training episodes, we record the average discounted return (ADR), which includes the costs associated with the line losses, tap changes, and the penalty of voltage violations.

The total operation cost (OC), the number of tap changes (# of TC), the number of voltage violations (# of VV), and the accumulated per unit voltage violation (AVV) over the test week are recorded in Table 4.1 for all the reinforcement algorithms and the benchmark algorithms. The operation cost includes the costs associated with the line losses and the tap changes. The accumulated per unit voltage violation is calculated as $\sum_i^N \sum_t [\max(0, |v_i^t| -$

$\overline{v}) + \max(0, \underline{v} - |v_i^t|)]$.

Table 4.1: Performance comparison of Volt-VAR control algorithms

	Algorithm	OC ($)	# of TC	# of VV	AVV (per unit)
4-bus test case	Baseline	150.13	0	91	2.748
	MPC (Actual)	111.44	18	0	0
	MPC (Forecast)	111.89	20	0	0
	CPO	115.01	9	5	0.044
	TRPO	120.05	3	16	0.286
13-bus test case	Baseline	77.88	0	268	2.673
	MPC (Actual)	58.05	6	0	0
	MPC (Forecast)	58.44	6	0	0
	CPO	58.92	6	0	0
	TRPO	61.29	3	2	0.004

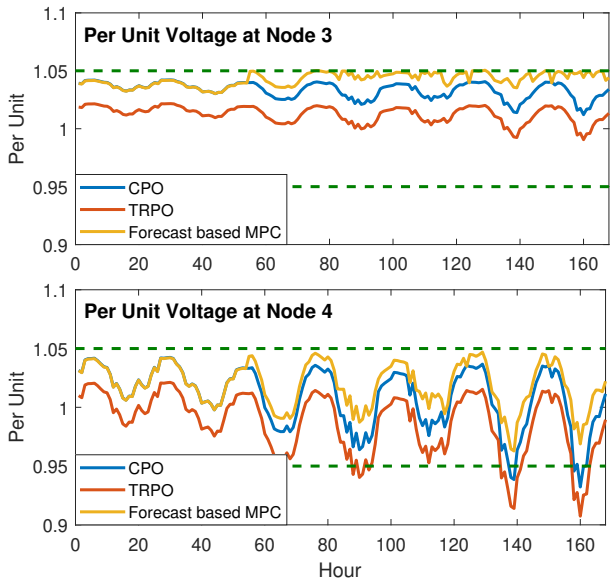

Figure 4.2: Comparison of voltage profiles on the 4-bus test feeder

The MPC with actual load represents the global optimal solution. As shown in Table 4.4, the CPO algorithm is capable of achieving a near-optimal operational cost and

is nearly constraint-satisfying. The CPO algorithm yields a lower operation cost compared to the TRPO algorithm. The per unit voltages at node 3 and 4 of the 4-bus test feeder are depicted in Fig. 4.2. It can be seen that the voltage solutions at node 3 of the MPC-based approach with forecasted load hit the upper bound a few times. This is common for optimization approaches as the optimal solutions are likely to be boundary points. By following the CPO algorithm, the voltage profiles at node 4 nearly stay in bounds all the time except for 5 minor violations. The CPO algorithm outperforms the TRPO algorithm by approximately satisfying the voltage constraints all the time.

Table 4.2: Computation time of Volt-VAR control algorithms

	Algorithm	Average Time (s)	Maximum Time (s)
4-bus test case	MPC (GUROBI)	10.43	90.28
	MPC (MOSEK)	346.80	3904.22
	TRPO/CPO	$< 10^{-3}$	$< 10^{-3}$
13-bus test case	MPC (GUROBI)	4.69	8.57
	MPC (MOSEK)	53.83	328.98
	TRPO/CPO	$< 10^{-3}$	$< 10^{-3}$

The average and the maximum computation time of the MPC-based algorithms with different solvers and the policy gradient methods to determine the tap positions at each hour are provided in Table 4.2. Without parallel computing (MOSEK), the computation time of the MPC-based algorithm could exceed 1 hour in the worst case on an entry level DELL desktop. On the other hand, once trained the policy gradient methods have a much faster execution speed, which makes them suitable for online applications. Moreover, the MPC-based algorithms require accurate and complete topology model and parameters of the distribution network, which are not often available.

4.2.5 Conclusion

In this section, the Volt-VAR control problem is modeled as a CMDP and solved with policy gradient methods for the first time. The constrained policy optimization algorithm is adopted to enable safe exploration for the controller. Both policy and state-value functions are approximated by neural networks. The structure of the policy network is tailored to achieve better scalability for the Volt-VAR control problem. The performance of the policy gradient methods and benchmarking algorithms are validated with the IEEE 4-bus and 13-bus test feeders. The results show that the constrained policy optimization algorithm can achieve near-optimal solutions with negligible voltage violations. Compared to the conventional optimization based approach, the proposed reinforcement learning algorithm is better suited for online VVC tasks where accurate and complete distribution network models are not available.

4.3 Volt-VAR Control with Off-policy Deep Reinforcement Learning Algorithm

4.3.1 Problem Formulation

In this section, we first introduce the preliminaries for CMDP and then formulate the VVC problem as a CMDP.

A. Preliminaries of Constrained Markov Decision Process

As a formalization of sequential decision making, CMDP is defined by a tuple of a state space $\mathcal{S}$, an action space $\mathcal{A}$, a reward function R, a cost function R^c, a transition

probability function Pr, and a discount factor $\gamma \in (0, 1)$.

In a CMDP, a learner and decision maker, also called an agent, interacts with the environment at each of a sequence of discrete time steps, $t = 0, 1, 2, 3, \ldots, T$. At each time step t, the agent observes the state of the environment $\boldsymbol{s}_t \in \mathcal{S}$ and selects an action $\boldsymbol{a}_t \in \mathcal{A}$. One time step later, the agent receives a numerical reward $R(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}) \subset \mathbb{R}$ and a numerical cost $R^c(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}) \subset \mathbb{R}$. The state of the environment becomes $\boldsymbol{s}_{t+1}$ according to the transition probability function $Pr(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t, \boldsymbol{a}_t)$.

The goal of the agent is to find a control policy π that maximizes the expected discounted return with respect to reward function J subject to a budget constraint for the expected discounted return with respect to cost function J^c:

$$\max_{\pi} J(\pi) \quad s.t. \quad J^c(\pi) \leq \overline{J} \tag{4.26}$$

where π is a mapping from a state space $\mathcal{S}$ to a action space $\mathcal{A}$ for a deterministic policy or a mapping from states to probabilities of selecting different actions for a probabilistic policy. The expected discounted return of policy π with respect to the reward is defined as: $J(\pi) = \underset{\tau \sim \pi}{E}[\sum_{t=0}^{T} \gamma^t R_t]$, where τ is a trajectory or sequence of states and actions, $\{\boldsymbol{s}_0, \boldsymbol{a}_0, \boldsymbol{s}_1, \boldsymbol{a}_1, ..., \boldsymbol{s}_{T-1}, \boldsymbol{a}_{T-1}, \boldsymbol{s}_t\}$. R_t is the short name for $R(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1})$. Similarly, the expected discounted return of policy π with respect to cost function is defined as $J^c(\pi) = \underset{\tau \sim \pi}{E}[\sum_{t=0}^{T} \gamma^t R_t^c]$, where R_t^c is $R^c(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1})$ for short.

Finally, we define two important value functions, state-value function $V^\pi(\boldsymbol{s})$ and action-value function $Q^\pi(\boldsymbol{s}, \boldsymbol{a})$, as follows:

$$V^\pi(\boldsymbol{s}) = \underset{\tau \sim \pi}{E}[\sum_{t=0}^{T} \gamma^t R_t | \boldsymbol{s}_0 = \boldsymbol{s}] \tag{4.27}$$

$$Q^{\pi}(\boldsymbol{s}, \boldsymbol{a}) = \underset{\tau \sim \pi}{E}[\sum_{t=0}^{T} \gamma^{t} R_t | \boldsymbol{s}_0 = \boldsymbol{s}, \boldsymbol{a}_0 = \boldsymbol{a}] \tag{4.28}$$

$V^{\pi}(\boldsymbol{s})$ represents the expected discounted return starting from state $\boldsymbol{s}$ and taking actions following policy π thereafter. $Q^{\pi}(\boldsymbol{s}, \boldsymbol{a})$ represents the expected discounted return starting from state $\boldsymbol{s}$, taking action $\boldsymbol{a}$, and thereafter following policy π. The value functions satisfy the Bellman equations:

$$V^{\pi}(\boldsymbol{s}_t) = \underset{\substack{a_t \sim \pi \\ \boldsymbol{s}_{t+1} \sim Pr}}{E} [R_t + \gamma V^{\pi}(\boldsymbol{s}_{t+1})] \tag{4.29}$$

$$Q^{\pi}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \underset{\substack{\boldsymbol{a}_{t+1} \sim \pi \\ \boldsymbol{s}_{t+1} \sim Pr}}{E} \Big[R_t + \gamma Q^{\pi}(\boldsymbol{s}_{t+1}, \boldsymbol{a}_{t+1}) \Big] \tag{4.30}$$

B. Formulating VVC Problem as a CMDP

Similarly as in section , the distribution system operator or controller is treated as the agent who interacts with the distribution grid. The primary controllable devices for the VVC task are selected to be voltage regulators, on-load tap changers, and switchable capacitor banks. The state of the environment is defined as $\boldsymbol{s} = (\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{Tap}, t)$. $\boldsymbol{P}$ and $\boldsymbol{Q}$ are the vectors of nodal real and reactive power injections. $\boldsymbol{Tap}$ is the vector of the current tap/on-off positions of controllable devices. The action taken by a VVC agent at each time step is changing the tap/on-off positions of controllable devices to $\boldsymbol{Tap}'$. The size of the action space is $\Pi_{i=1}^{N_c} |\mathcal{A}_i|$, where N_c is the number of controllable devices and $|\mathcal{A}_i|$ denotes the number of tap/on-off positions of device i.

The VVC agent aims at reducing the distribution network losses and the operating costs of the controllable devices. Thus, the reward function R_t of the VVC agent can be

defined as the negative of the total operational costs, which includes the cost of real power losses and the device switching cost:

$$R_t = -\Big[C_e P_{loss}(t) + \sum_{j=1}^{N_c} C_j^T |Tap_j(t+1) - Tap_j(t)|\Big] \tag{4.31}$$

The switching cost of a device is calculated as the product of the absolute change in tap positions between consecutive time steps and the per tap position change cost C_j^T for device j. C_e and $P_{loss}(t)$ denote the electricity price and the total real power loss at time step t respectively. The total real power loss is defined as the summation of real power losses of all lines and devices in the distribution network.

To maintain nodal voltage profiles within a desirable range, the cost function is chosen as the number of voltage constraint violations across all the nodes:

$$R_t^c = \sum_{i=1}^{N} [\mathbb{1}(|V_i^{t+1}| > \overline{V}) + \mathbb{1}(|V_i^{t+1}| < \underline{V})] \tag{4.32}$$

where $\mathbb{1}(\cdot)$ is the indicator function. V_i^{t+1} is the voltage of node i at hour $t+1$; $\overline{V}$ and $\underline{V}$ are the upper and lower limits for voltage magnitudes. N is the total number of nodes.

By evaluating the feedback in the form of rewards and costs defined above via past and/or future interactions with the physical environment, the VVC agent tries to learn a control policy that minimizes the total operational cost while satisfying the voltage constraints.

4.3.2 Safe Off-policy Deep Reinforcement Learning Algorithm

In this section, we develop an innovative deep RL algorithm named constrained soft actor-critic (CSAC) to solve the VVC problem, which is formulated as a CMDP. A

suitable RL algorithm for solving the VVC problem should be sample efficient, scalable, and safe to implement in the real world.

Sample efficiency: Unlike the domain of computer games, we can not repeatedly generate a tremendous amount of operation experiences for VVC in real world distribution feeders with low cost. Thus, it is crucial for us to develop off-policy RL algorithms, where the learned control policy (target policy) and the policy that generated control behaviors (behavior policy) are different. Being able to reuse the historical operational experiences, the off-policy RL algorithms are much more sample efficient than the on-policy ones.

Scalability: In a VVC problem, the network loss is determined by the tap positions of all controllable devices together. The number of feasible control actions increases exponentially with the number of controllable devices. Thus, in order to solve a large-scale VVC problem, it is important to learn a control policy whose number of parameters increases approximately linearly with the number of controllable devices.

Constraint satisfaction: In RL, agents are often given complete freedom to learn a control policy by trial and error. However, in a real-world VVC problem, this is unacceptable. Certain exploratory control actions may lead to significant voltage violations in the distribution network causing equipment damage and undermining the reliability of the network. Thus, we want to develop a RL algorithm, which can achieve near constraint satisfaction at all times.

In the following subsections, we first introduce the actor-critic method, which is a widely used policy gradient method. Next, we present the state-of-the-art maximum-entropy based off-policy RL algorithm, soft actor-critic (SAC). We then propose an inno-

vative off-policy RL algorithm called CSAC to solve the VVC problem. This is followed by a presentation of the detailed algorithm design for CSAC. At last, we derive the policy gradient for discrete actions and describe the device-decoupled policy network structure and ordinal encoding for discrete actions.

A. Actor-Critic Method

The basic policy gradient method is an actor-only method, where the actor refers to the policy function. Actor-only methods typically learn parameters for the approximated policy function based on episodic gains from Monte-Carlo sample trajectories. This often leads to high variance and slow learning [139]. To overcome these shortcomings, the actor-critic method is proposed to update policy function parameters based on the approximated value function that is a synonym for the critic. The iterative framework for a typical actor-critic method is shown in Algorithm 6. At each iteration, the actor first generates samples by taking actions according to the current policy. Then, the critic evaluates the quality of the current policy by adjusting the value function estimates based on the temporal difference [139] according to (4.30). At last, the actor is updated by using the information from the critic.

B. Soft Actor-Critic

The commonly used actor-critic algorithms such as PPO [144] and A3C [145] are notoriously sample inefficient, because they require new samples to be generated according to the latest policy at each gradient step. Although off-policy policy gradient algorithms

Algorithm 6 Actor-Critic Algorithm

1: Initialize policy and value function parameters

2: **repeat**

3: Generate samples by taking actions according to the current policy

4: Update value function parameters according to (4.30)

5: Update policy parameters based on value function

6: **until** converge

such as DDPG [146] were introduced to improve sample efficiency, they are often brittle with respect to their hyperparameters. To address these challenges, the off-policy maximum-entropy deep RL algorithm, SAC [30], is developed to provide a robust and sample-efficient learning, which achieves the state-of-the-art performance.

The SAC is built on the maximum-entropy RL framework [147, 148], which maximizes not only the expected return but also the entropy of the policy. The entropy for a probabilistic policy at state $\boldsymbol{s}_t$ is defined as $H(\pi(\cdot|\boldsymbol{s}_t)) = -\sum_{\boldsymbol{a}} \pi(\boldsymbol{a}|\boldsymbol{s}_t) \ln \pi(a|\boldsymbol{s}_t)$.

In the maximum-entropy RL framework, we typically work with the regularized value functions [149] defined as:

$$V_h^\pi(\boldsymbol{s}) = \underset{\tau\sim\pi}{E}\Big[\sum_{t=0}^{T} \gamma^t \big(R_t + \alpha H(\pi(\cdot|\boldsymbol{s}_t))\big)\big|\boldsymbol{s}_0 = s\Big] \tag{4.33}$$

$$Q_h^\pi(\boldsymbol{s}, \boldsymbol{a}) = \underset{\tau\sim\pi}{E}\Big[\sum_{t=0}^{T} \gamma^t R_t + \alpha \sum_{t=1}^{T} \gamma^t H(\pi(\cdot|\boldsymbol{s}_t))\big|\boldsymbol{s}_0 = s, \boldsymbol{a}_0 = a\Big] \tag{4.34}$$

The corresponding entropy-regularized Bellman equations are:

$$V_h^\pi(\boldsymbol{s}_t) = \underset{\substack{a_t\sim\pi \\ \boldsymbol{s}_{t+1}\sim Pr}}{E}\Big[R_t + \alpha H(\pi(\cdot|\boldsymbol{s}_t)) + \gamma V_h^\pi(\boldsymbol{s}_{t+1})\Big] \tag{4.35}$$

$$Q_h^\pi(\boldsymbol{s}_t, \boldsymbol{a}_t) = \underset{\substack{\boldsymbol{a}_{t+1}\sim\pi \\ \boldsymbol{s}_{t+1}\sim Pr}}{E}\Big[R_t + \gamma\big(Q_h^\pi(\boldsymbol{s}_{t+1}, \boldsymbol{a}_{t+1}) + \alpha H(\pi(\cdot|\boldsymbol{s}_{t+1}))\big)\Big] \tag{4.36}$$

The two regularized value functions have the following relationship:

$$V_h^{\pi}(\boldsymbol{s}_t) = \underset{\boldsymbol{a}_t \sim \pi}{E}[Q_h^{\pi}(\boldsymbol{s}_t, \boldsymbol{a}_t)] + \alpha H(\pi(\cdot|\boldsymbol{s}_t)) \tag{4.37}$$

Equation (4.37) allows us to derive the closed-form solution [149] of the policy $\pi^{\dagger}(\cdot|s) = arg\max_{\pi \in \Delta}\{V_h^{\pi}(\boldsymbol{s})\}$, where $\Delta = \{\pi | \pi \geq 0, \mathbf{1} \cdot \pi = 1\}$, as:

$$\pi^{\dagger}(\cdot|\boldsymbol{s}) = \frac{e^{Q_h^{\pi}(\boldsymbol{s},\cdot)/\alpha}}{\sum_{\boldsymbol{a}} e^{Q_h^{\pi}(\boldsymbol{s},\boldsymbol{a})/\alpha}} \tag{4.38}$$

When Q_h^{π} converges to Q_h^*, the optimal policy $\pi^*(\cdot|\boldsymbol{s})$ also achieves optimal value $V_h^*(\boldsymbol{s})$ for all states $\boldsymbol{s}$. By using the closed-form solution, the updating schema of Q-function could be realized in an off-policy fashion.

Algorithm 7 Soft Actor-Critic

1: Initialize policy and regularized value function parameters
2: **repeat**
3: Sample from data buffer
4: Update parameters of value functions according to (4.36)
5: Update policy parameters according to (4.38)
6: **until** converge

The overall framework of SAC is summarized in Algorithm 7. The implementation details such as the clipped double-Q learning [150], the baseline value function [139], and the delayed update of value function [30] are omitted here.

C. Constrained Soft Actor-Critic

Although SAC has been successfully demonstrated on a range of challenging control tasks, it is designed to solve MDPs and cannot handle CMDPs with physical constraints.

If one simply augments the reward with the product of a fixed penalty factor and constraint violation, then the learned policy will be either too conservative or infeasible. In this subsection, we propose CSAC by extending SAC algorithm to satisfy the operational constraints in CMDPs.

The goal of the SAC algorithm is to find an optimal policy, which maximizes the regularized state-value function, $\max_\pi \underset{\boldsymbol{s}\sim D}{E}[V_h^\pi(\boldsymbol{s})]$, where D is the historical operation data buffer, i.e., the set of experience tuple $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}, R_t, R_t^c)$.

Moreover, in real-world control problems, it is necessary to enforce operational constraints. For the VVC problem, we need to limit the number of total voltage constraint violations at each time step, i.e., $R_t^c \leq \overline{R^c}$. R_t^c is defined in (4.32), and $\overline{R^c}$ is the upper bound. For a finite horizon CMDP, the corresponding limit $\overline{V^c}$ for the state-value function associated with the operation constraint can be set as $V^{c,\pi}(\boldsymbol{s}) \leq \overline{V^c} = (1-\gamma^T)/(1-\gamma)\overline{R^c}$, where T is the episode length. Note that other types of operational constraints can be enforced in a similar manner.

Within the maximum-entropy RL framework, the optimal policy of CMDP can be obtained by solving:

$$\begin{aligned} &\max_\pi \underset{\boldsymbol{s}\sim D}{E}[V_h^\pi(\boldsymbol{s})] \\ s.t. \quad &\underset{\boldsymbol{s}\sim D}{E}[V^{c,\pi}(\boldsymbol{s})] \leq \overline{V^c} \end{aligned} \tag{4.39}$$

The Lagrange function of the constrained optimization problem can be written as:

$$\mathcal{L}(\pi, \lambda) = \underset{\boldsymbol{s}\sim D}{E}[V_h^\pi(\boldsymbol{s})] + \lambda(\overline{V^c} - \underset{\boldsymbol{s}\sim D}{E}[V^{c,\pi}(\boldsymbol{s})]) = \underset{\boldsymbol{s}\sim D}{E}[V_h^{l,\pi}(\boldsymbol{s})] + \lambda\overline{V^c} \tag{4.40}$$

where

$$V_h^{l,\pi}(\boldsymbol{s}) = \underset{\tau\sim\pi}{E}\Big[\sum_{t=0}^{T}\gamma^t\big(R(\boldsymbol{s}_t,\boldsymbol{a}_t,\boldsymbol{s}_{t+1}) - \lambda R^c(\boldsymbol{s}_t,\boldsymbol{a}_t,\boldsymbol{s}_{t+1}) + \alpha H(\pi(\cdot|\boldsymbol{s}_t)\big)\big|\boldsymbol{s}_0 = \boldsymbol{s}\Big] \tag{4.41}$$

The method of multipliers can be used to solve the constrained optimization problem. At k-th iteration , given a multiplier $\lambda^k \geq 0$, we can maximize $\mathcal{L}(\cdot,\lambda^k)$, over policy domain thereby obtaining a policy π^k. We then set

$$\lambda^{k+1} = [\lambda^k - \delta_\lambda \nabla_\lambda \mathcal{L}]^+ = [\lambda^k + \delta_\lambda(\underset{\boldsymbol{s}\sim D}{E}[V^{c,\pi_k}(\boldsymbol{s})] - \overline{V^c})]^+ \tag{4.42}$$

and repeat the process. δ_λ is the step size for the λ update process. $[]^+$ is the projection to non-negative real numbers.

With a small α and $H(\pi) \approx 0$ at convergence, $V_h^{c,\pi}(\boldsymbol{s}) \approx V^{c,\pi}(\boldsymbol{s})$. To have consistent forms of value functions, the update of the Lagrange multiplier can be redesigned as:

$$\lambda^{k+1} = [\lambda^k + \delta_\lambda(\underset{\boldsymbol{s}\sim D}{E}[V_h^{c,\pi_k}(\boldsymbol{s})] - \overline{V^c})]^+ \tag{4.43}$$

where $V_h^{c,\pi_k}(\boldsymbol{s})$ is the state-value function associated with the operation constraint at k-th iteration.

It has been shown that the iterative approach for updating the parameters of control policy and Lagrange multiplier will guarantee the convergence to a local optimal and feasible solution when the following three assumptions hold [151, 152]. First, $V_h^{\pi}(\boldsymbol{s})$ is bounded for all policies $\pi \in \Pi$. Second, every local minima of $J^c(\pi)$ is a feasible solution. Third, $\sum_{k=0}^{\infty}\delta_\theta = \sum_{k=0}^{\infty}\delta_\lambda = \infty$, $\sum_{k=0}^{\infty}\delta_\theta^2 + \sum_{k=0}^{\infty}\delta_\lambda^2 < \infty$, and $\lim_{k\to\infty}\delta_\lambda/\delta_\theta = 0$. δ_θ is the step size for updating the parameters θ of the policy neural network.

Note that for finite episodic cases, δ_λ can be set to be smaller than δ_θ in practice.

If the local optimal solution is not feasible, then the algorithm can be restarted with a larger initial value for λ.

D. Algorithm Design for CSAC

The proposed CSAC is an off-policy RL algorithm, which allows the offline training of control policy in an iterative manner. The overall framework of the CSAC is summarized in Algorithm 8. In each iteration, we first perform stochastic gradient descent to update the parameters of neural networks, which approximate the value functions and policy function. Then, we update the Lagrange multiplier of the constrained optimization problem as shown in (4.43).

Two sets of neural networks are used to approximate the action-value functions Q_ψ and state-value functions V_ϕ. The first set of value functions, parameterized with ψ^l and ϕ^l, are associated with the value functions in the Lagrange function (4.40). The second set of value functions, parameterized with ψ^c and ϕ^c, are associated with the constraint. The policy function is approximated by a neural network π_θ parameterized by θ.

The parameters of the action-value neural networks, Q_ψ, are updated by minimizing the mean-square-error (MSE), $1/|B|\sum_B(Q_\psi - \hat{Q})^2$, where B is a randomly selected mini-batch of samples, i.e., a set of transition tuples $\{(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}, R_t, R_t^c)\}$. $|B|$ denotes the size of the min-batch. The training target $\hat{Q}$ is calculated as $\hat{Q}(\boldsymbol{s}_t, \boldsymbol{a}_t) = r_t + \gamma V_\psi(\boldsymbol{s}_{t+1})$, where r_t is $R_t - \lambda R_t^c$ for the neural network associated with the Lagrange function and R_t^c for the neural network associated with the constraint. Similarly, the state-value networks, V_ϕ, are updated by minimizing the MSE, $1/|B|\sum_B(V_\phi - \hat{V})^2$, where the target $\hat{V}(\boldsymbol{s}_t) = Q_\psi(\boldsymbol{s}_t, \boldsymbol{a}_t) - \alpha \ln \pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t)$. The parameters of the policy neural network is up-

dated by minimizing the loss,

$$\frac{1}{|B|}\sum_{B}\ln\pi_\theta(\hat{\boldsymbol{a}}_t|\boldsymbol{s}_t)(\alpha\ln\pi_\theta(\hat{\boldsymbol{a}}_t|\boldsymbol{s}_t)-Q_{\psi^l}(\boldsymbol{s}_t,\hat{\boldsymbol{a}}_t)+V_{\phi^l}(\boldsymbol{s}_t)) \tag{4.44}$$

where $\hat{\boldsymbol{a}}_t$ is the sampled action from $\pi_\theta(\cdot|\boldsymbol{s}_t)$. The derivation for the policy gradient is provided in the subsection 4.3.2.

Algorithm 8 CSAC Algorithm

Initialize network parameters and Lagrange multiplier λ
repeat
  for each sample step **do**
    $\boldsymbol{a}_t \sim \pi(\cdot|\boldsymbol{s}_t)$
    $D \leftarrow D \cup (\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}, R_t, R_t^c)$
  end for
  for each gradient step with sample batch B **do**
    Update action value networks Q_ψ
    Update state value networks V_ϕ
    Update policy network π_θ
    $\lambda \leftarrow [\lambda + \delta_\lambda \sum_B (V_{\phi^c} - \overline{V^c})/|B|]^+$
  end for
until converge

The neural networks approximating V and Q functions use the state vector s and the state action pair $\boldsymbol{s}, \boldsymbol{a}$ as inputs, where $\boldsymbol{a}$ is treated as a vector of ordinal variables. The outputs of these two networks are the corresponding target state and action values. The policy network has a special design, which will be described in subsection 4.3.2.

In order to stabilize the training process, the delayed update of value function [30] is adopted in our algorithm. The training labels for Q networks are modified as $\hat{Q}(\boldsymbol{s}_t, \boldsymbol{a}_t) = r_t + \gamma V_{\psi_{targ}}(\boldsymbol{s}_{t+1})$, where $V_{\psi_{targ}}$ are the extra copies of V networks, whose parameter ψ_{targ} updates are delayed at each gradient step by $\phi_{targ} = (1-\rho)\phi_{targ} + \rho\phi$, where $\rho \in (0, 1)$. To mitigate the positive bias in the policy update step, the clipped double Q-learning technique [150] is adopted. The training labels for V networks are modified as $\hat{V}(\boldsymbol{s}_t) = \min_{i=1,2} Q_{\psi_i}(\boldsymbol{s}_t, \boldsymbol{a}_t) - \alpha \ln \pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t)$, where two sets of Q networks, $Q_{\psi_1} = \{Q_{\psi_1^l}, Q_{\psi_1^c}\}, Q_{\psi_2} = \{Q_{\psi_2^l}, Q_{\psi_2^c}\}$, are maintained and trained separately.

E. Policy Gradient for Discrete Action

Discrete control variables are needed to represent the control actions in the VVC problem such as changing the tap/on-off positions of voltage regulators, on-load tap changers, and switchable capacitor banks. The policy gradient of the SAC algorithm designed for a continuous control problem can not be directly applied for our proposed CSAC. Specifically, in SAC, the sampled actions are reparameterized with $\hat{\boldsymbol{a}}_\theta = \mu_\theta + v_\theta \mathcal{N}(0, 1)$, where μ_θ and v_θ are the outputs of mean values and variances from the Gaussian policy network. $\mathcal{N}(0, 1)$ is the standard normal distribution. Therefore, the $\hat{\boldsymbol{a}}_\theta$ is differentiable with respect to θ. However, it is no longer true for the discrete actions which are sampled with the output distribution of the policy network. For discrete action space, the policy gradient can be derived in a similar fashion to the policy gradient theorem [139, 153] to maximize the state-value function:

$$\nabla_\theta V_h^\pi(\boldsymbol{s}) = \nabla_\theta \sum_a \pi_\theta(\boldsymbol{a}|\boldsymbol{s})\big(Q_h^\pi(\boldsymbol{s}, \boldsymbol{a}) - \alpha \ln \pi_\theta(\boldsymbol{a}|\boldsymbol{s})\big)$$

$$= \underset{a\sim\pi_\theta}{E}[\nabla_\theta \ln \pi_\theta(\boldsymbol{a}|\boldsymbol{s})(Q_h^\pi(\boldsymbol{s},\boldsymbol{a}) - \alpha \ln \pi_\theta(\boldsymbol{a}|\boldsymbol{s}))]$$

$$= \underset{a\sim\pi_\theta}{E}\Big[\nabla_\theta \ln \pi_\theta(\boldsymbol{a}|\boldsymbol{s})\big(Q_h^\pi(\boldsymbol{s},\boldsymbol{a}) - V_h^\pi(\boldsymbol{s}) - \alpha \ln \pi_\theta(\boldsymbol{a}|\boldsymbol{s})\big)\Big] \tag{4.45}$$

The regularity condition, $\sum_a \pi_\theta(\boldsymbol{a}|\boldsymbol{s})\nabla_\theta \ln \pi_\theta(\boldsymbol{a}|\boldsymbol{s}) = 0$, is used for the derivation of the second line. Note that the loss function for updating the parameters θ of the policy neural network is chosen as (4.44), whose partial derivative is the negative of (4.45).

F. Device-Decoupled Policy Network Structure and Ordinal Encoding for Discrete Actions

Since only a single tap position can be chosen by each of the remotely controllable devices for VVC problems, we design the policy neural network with a device-decoupled structure. The input of the policy neural network is the state vector s and the outputs are the probabilities of selecting a tap position for each of the N_c devices. Thus, the dimensionality of the output layer is $\sum_{i=1}^{N_c} |\mathcal{A}_i|$, where $|\mathcal{A}_i|$ denotes the number of tap positions for device i. In this way, the network size only increases linearly with N_c. The j-th action of the i-th device corresponds to the logit output l_{ij} of the last hidden layer of the neural network. The probability p_{ij} of choosing j-th action for the i-th device can be calculated by combining $l_{ij}, 1 \leq j \leq |\mathcal{A}_i|$ via a softmax function, $p_{ij} = exp(l_{ij})/\sum_j exp(l_{ij})$. The final probability of a tap position combination of all the devices is equal to the product of the probability of each individual device taking its own action, $p(\boldsymbol{a}) = \Pi_{i=1}^{N_c} p_i(a_i)$, where $\boldsymbol{a}$ is the vector of chosen actions across all the devices and a_i is the chosen action of i-th device.

Note that the discrete controls actions of each remotely controllable device can

be represented by an ordinal variable. For example, the control actions of an on-load tap changer with 3 tap positions that correspond to turns ratios of 0.95, 1, and 1.05 can be deemed as a discretization of an ordinal variable of turns ratio. Thus, we adopt an ordinal representation [154] for all the discrete actions of a device to encode the natural ordering between the discrete actions.

Specifically, each subset of the logit outputs corresponding to a device is first pre-processed as follows:

$$l'_{ij} = \sum_{m \leq j} \ln o_{im} + \sum_{m > j} \ln(1 - o_{im}),\ i = 1, 2..., N_c \tag{4.46}$$

where the sigmoid function is first applied to the logits, $o_{ij} = sigmoid(l_{ij})$, and l'_{ij} is the transformed logit after the ordinal encoding. Then the probability of device i taking action j can be calculated via $p'_{ij} = exp(l'_{ij}) / \sum_j exp(l'_{ij})$.

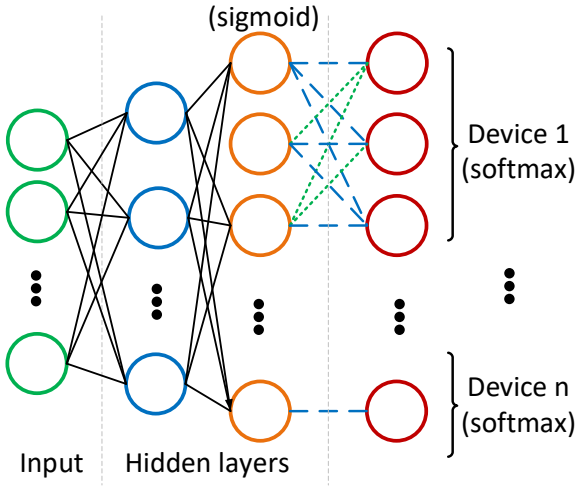

Figure 4.3: Device-decoupled structure of the policy neural network

The device-decoupled structure of the policy neural network is depicted in Fig.4.3, where the blue lines denote the connections associated with $\ln(x)$ and the yellow lines

denote the connections associated with $\ln(1-x)$. Note that (4.46) is equivalent to encoding the j-th action of a control device as a vector, $[1, ...1, 0...]$, where the first j elements are set as 1s and the rest of the elements are set as 0s. By introducing an inductive bias which appropriately distinguishes the dissimilarity among the discrete actions, the ordinal encoding further improves the learning efficiency of our proposed CSAC algorithm.

4.3.3 Numerical Study

Numerical studies are carried out on distribution test feeders to validate the sampling efficiency, scalability, optimality, and safety of the proposed CSAC algorithm for solving VVC problems. We also performed a comprehensive comparison between the proposed algorithm and four benchmark algorithms including three RL algorithms and two optimization-based algorithms.

A. Simulation Setup

The IEEE 4-bus, 34-bus and 123-bus distribution test feeders [75] are used in the numerical simulations. In the 4-bus feeder, a voltage regulator is located at node 1 and an on-load tap changer connects node 2 and 3. We add a capacitor bank with 200 kVar rating to node 4. In the 34-bus test feeder, a voltage regulator is at node 800. There are two transformers connecting node 814 to node 850 and node 852 to node 832 respectively. Two capacitors are placed at node 844 (100 kVar) and node 847 (150 kVar). In the 123-bus test feeder, a voltage regulator is at node 150. There are three on-load tap changers, which connect node 10 to node 15, node 160 to node 67, and node 25 to node 26 respectively. Four capacitors are placed at node 83 (200 kVar), node 88 (50 kVar), node 90 (50 kVar), and

node 92 (50 kVar). All voltage regulators and on-load tap changers have 11 tap positions, which correspond to turns ratios ranging from 0.95 to 1.05. The capacitors can be switched on/off remotely and the number of 'tap positions' is treated as 2.

In the initial state, the turns ratios of voltage regulators and on-load tap changers are 1 and the capacitors are switched off. The electricity price C_e is assumed to be $\$40/MWh$. The operating cost per tap change C_j^T is set to be \$0.1 for all devices. One year of hourly smart meter energy consumption data [155] from London is used. The aggregated load data is scaled and allocated to each node according to the existing spatial load distribution of the IEEE standard test cases. 10 weeks of randomly selected data are used for out-of-sample testing. The rest of the data are used for training purposes. For DRL approaches, the reward and the cost are derived based on the line losses and nodal voltages calculated from the power flow simulations. For the three IEEE distribution test feeders, when the nodal voltages are within appropriate bounds, the line flow limits are also satisfied. Thus, only the voltage constraints are explicitly stated in the problem formulation. The upper limit for the number of voltage violations $\overline{V^c}$ is set as 0. The parameter settings for the reinforcement learning algorithms are provided in Table 4.3 below.

B. Setup of the Benchmark and Our Proposed Algorithms

The deep Q-network (DQN) [31] algorithm, an extension of the tabular Q-learning for the VVC [136], is chosen as the first benchmark RL algorithm. DQN algorithm is one of the most widely used off-policy RL algorithms for solving MDP. In order to apply DQN for CMDP, a penalty term for the voltage violation is added to the reward function as

Table 4.3: Parameter settings for reinforcement learning algorithms

Parameters	4-bus	34-bus	123-bus
Size of Hidden Layers	(64, 32)		
Activation Function of Hidden Layers	relu		
Batch Size	256		
Initial Value of λ	0		
Discount Factor γ	0.99		
Temperature Parameter α	0.02	0.02	0.05
Step Size for Q Networks δ_ψ	1e-3		
Step Size for V Networks δ_ϕ	5e-4		
Step Size for π Network δ_θ	1e-3		
Step Size for λ Update δ_λ	1e-5		
Delay Factor ρ	5e-4		

$R_t - C_V R_t^c$, where the penalty coefficient C_V is set as \$1 per voltage violation per node. Constrained Policy Optimization (CPO) algorithm, a state-of-the-art RL algorithm for solving CMDP, is chosen as the second benchmark RL algorithm. CPO not only guarantees monotonic policy improvement at each policy iteration step but also ensures constraint satisfaction throughout the training process given that a feasible policy is recovered.

Both our proposed CSAC and the DQN algorithm are off-policy RL algorithms. A single sample is collected at each training step for these two algorithms. On-policy RL algorithms such as CPO typically require a large number of new samples to be collected in order to accurately estimate the state values. In this study, the sampling size of each training step of CPO is set to be 5000, which is determined by gradually increasing the sampling size until the algorithm can achieve a stable and reasonable performance. The length of each episode is set as a week, i.e., 168 hours. The weights of the neural networks are randomly initialized and updated with batch training. The batch size is set as 256.

To illustrate the effectiveness of proposed CSAC method, the SAC algorithm with fixed penalty coefficients is chosen as the third DRL benchmark. Except for removing the update step for λ, the same parameters are chosen as that of the CSAC algorithm. The parameters of neural networks are fine-tuned based on the training performance.

Two benchmark optimization-based algorithms for VVC problems are also implemented. The first benchmark optimization algorithm is implemented based on the single period (one hour) mixed-integer conic programming (MICP), which is the same as the discrete control stage without the chance constraints in [141]. Essentially, a multi-period VVC problem is solved for one hour at a time with the MICP algorithm. The second benchmark optimization algorithm is implemented by extending the single period MICP to multiple periods with model predictive control (MPC) framework as in [124] over a planning horizon of 24 hours. Note that for the optimization-based benchmarks, the actual future load is assumed to be given. The commercial solver GUROBI is used for both benchmark optimization algorithms.

C. Sample Efficiency

Evaluated based on the necessary number of samples to reach a stable solution, the sample efficiency of the proposed CSAC algorithm and the two benchmark RL algorithms is analyzed for the three distribution test feeders in this subsection. The number of training samples collected versus the average weekly return (AVR) on the testing weeks, i.e., the negative of the total operational costs associated with real power losses, tap changes, and voltage violations are shown in the top subfigures of Fig.4.4-4.6. The number of weekly voltage violations versus the number of training samples are shown in the bottom subfigures

of Fig.4.4-4.6. The solid curves are the average performances of 5 random experiments, and the light-colored regions represent the error bounds. As shown in Fig.4.4-4.6, to achieve

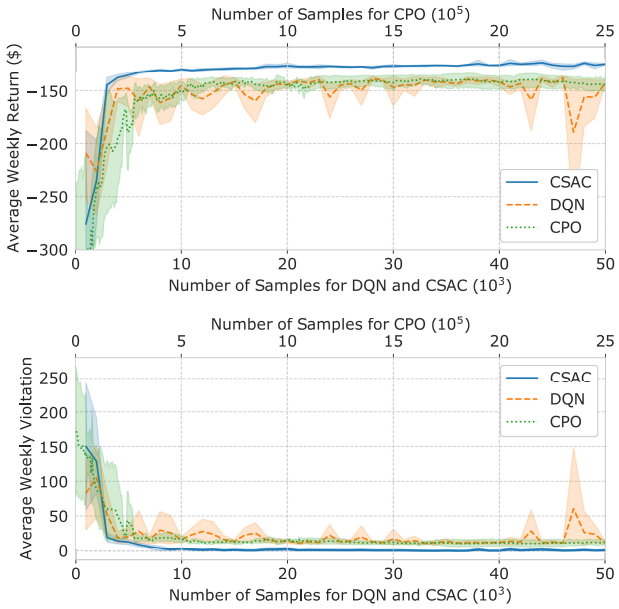

Figure 4.4: Average weekly return and voltage violation for 4-bus test feeder

the same level of performance, our proposed CSAC algorithm needs the least amount of training samples. The on-policy CPO algorithm needs a much higher number of training samples than the off-policy algorithms, CSAC and DQN. In the case of the 4-bus test feeder, CSAC and DNQ only need about 10,000 training samples to achieve stable performance, while CPO requires about 500,000 training samples to achieve stable performance.

The off-policy nature of CSAC algorithm not only significantly improves sample efficiency, but also allows us to reuse historical operational data. In contrast, the on-policy algorithms such as CPO need to generate new samples according to the latest policy at

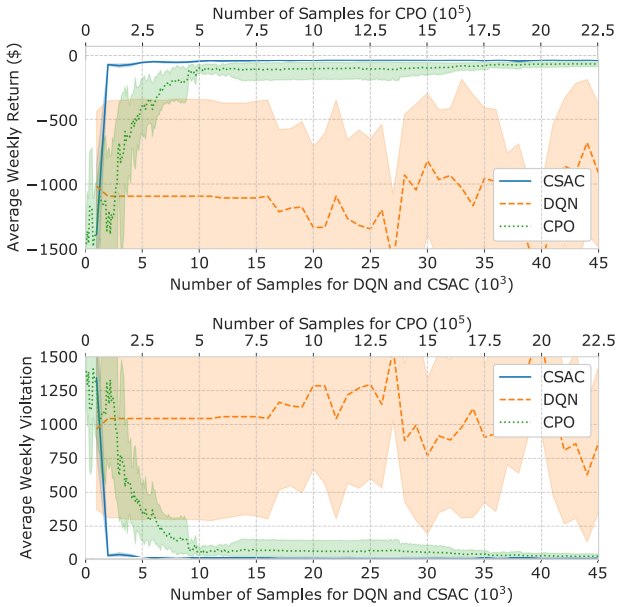

Figure 4.5: Average weekly return and voltage violation for 34-bus test feeder

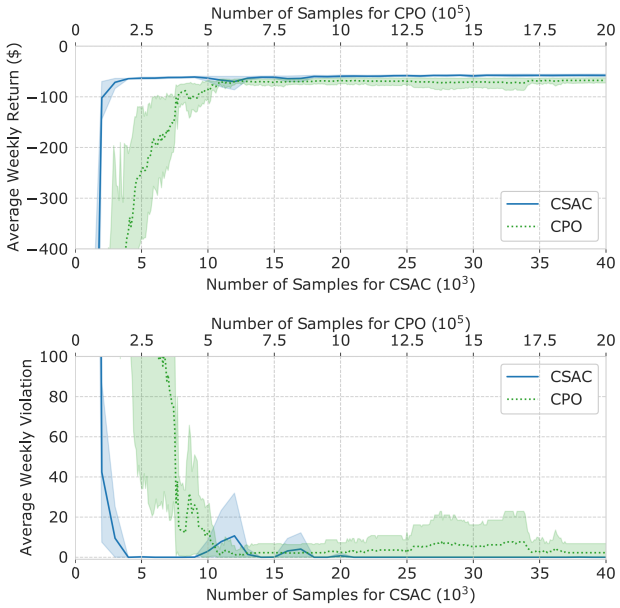

Figure 4.6: Average weekly return and voltage violation for 123-bus test feeder

every training step. Moreover, at each step of CPO, a large number of samples need to be collected to form an accurate estimate of the state values.

Table 4.4: Performance comparison of Volt-VAR control algorithms

Algorithm	AVR ($)			AVV		
	4-bus	34-bus	123-bus	4-bus	34-bus	123-bus
DQN	-140.22	-680.09	N/A†	10.6	630.40	N/A†
CPO	-139.27	-71.78	-68.88	9.6	27.34	1.12
CSAC	-126.58	-42.39	-57.43	0.18	0.06	0
MPC	-122.86	N/A*	N/A*	0	N/A*	N/A*
MICP	-133.26	-44.51	-66.99	0	0	0

* can not find a solution of a rolling step in 4 hours.

† can not finish one epoch of training in 10 hours.

D. Optimality, Constraint Satisfaction, and Scalability

The AVRs and the number of weekly average voltage constraints violations (AVVs) during the testing weeks of the proposed CSAC algorithm, the two benchmark RL algorithms, and the two benchmark optimization algorithms are shown in Table 4.4. The results of the RL algorithms are the averaged performances of 5 experiments, each with a different random seed.

As shown in the table, our proposed CSAC algorithm achieves the highest return among all algorithms for the 34-bus and 123-bus test feeders and the second highest return for the 4-bus test feeder. As the size of the distribution feeder increases, the advantage of our proposed CSAC algorithm becomes more pronounced. Furthermore, our proposed CSAC algorithm can satisfy the voltage constraints almost all the time whereas the other

benchmark RL algorithms may lead to significant voltage violations. Minor voltage violations do occur in the 4-bus and 34-bus test feeders when our proposed CSAC algorithm is used. However, the average voltage violation magnitude is much smaller than 0.01 per unit.

Although the MPC extension of the MICP algorithm achieves a better solution on the 4-bus test feeder, it is not scalable and can not find a solution of a single rolling step with 4 hours of computation time for both the 34-bus and the 123-bus systems. Similarly, the DQN algorithm is also not scalable and can not obtain a solution for the 123-bus system within a reasonable amount of time. This is because the number of Q values which need to be calculated for each greedy action selection, $\Pi_i^{N_c}|A_i|$, increases quickly with the number of controllable devices. In our proposed CSAC algorithm, the policy function can be approximated with a neural network whose size increases linearly with the number of devices as presented in 4.3.2. Therefore, our proposed device-decoupled encoding approach has much better scalability. Note that the same device-decoupled network structure was applied on the CPO algorithm, where the trust region constraint is enforced to limit the KL-divergence between the previous policy and the updated policy, $KL(\pi', \pi) = \sum_a \pi'(\boldsymbol{a}|\boldsymbol{s}) \ln(\pi'(\boldsymbol{a}|\boldsymbol{s})/\pi(\boldsymbol{a}|\boldsymbol{s})) \leq \delta$. The total KL-divergence can be decomposed with respect to each device as $KL(\pi', \pi) = \sum_i^{N_c} KL(\pi_i', \pi_i)$.

Table 4.5: Comparison between CSAC and SAC for Volt-VAR control

Algorithm	AVRwV ($)			AVV		
	4-bus	34-bus	123-bus	4-bus	34-bus	123-bus
CSAC	-126.40	-42.33	-57.43	0.18	0.06	0
SAC ($C_V = 0$)	-112.92	-36.50	-53.67	1020	1815.20	340.80
SAC ($C_V = 0.1$)	-125.49	-39.92	-58.19	69.33	10.62	7.94
SAC ($C_V = 1$)	-128.03	-45.31	-59.10	0.27	0.58	0

The comparison between SAC with different penalty coefficients and CSAC is performed to further demonstrate the effectiveness of the proposed method. The comparison results are summarized in 4.5. When the proposed CSAC is compared to the SAC with a constraint violation penalty factor $C_V = 1$, it is very clear that our proposed algorithm not only produces a higher weekly return without the penalty of voltage violations (AVRwV) and a smaller AVV.

4.3.4 Conclusion

A model-free DRL algorithm is proposed to solve the VVC problem without depending on accurate and complete distribution network topology and parameter information. The VVC problem is formulated as a CMDP and solved by our proposed CSAC algorithm, which is a safe off-policy DRL algorithm. In the algorithm implementation, the policy network is specially designed with a device-decoupled structure and an ordinal encoding scheme. Numerical studies conducted on the 4-bus, 34-bus, and 123-bus distribution test feeders demonstrate that the proposed algorithm achieves better sample-efficiency, scalability and constraint satisfaction than the state-of-the-art reinforcement learning algorithms and the conventional optimization-based algorithms.

Chapter 5

Algorithmic Trading with Virtual Bids

The regional wholesale electricity markets in the United States adopt the two-settlement structure, which includes the day-ahead (DA) market and the real-time (RT) market. The DA market is a forward market that determines the hourly DA locational marginal prices (LMPs), the unit commitment plans, and the DA dispatch schedules for generations and dispatchable loads. The decisions are made based on the supply offers and the demand bids submitted by load serving entities (LSEs), independent power producers (IPPs), and proprietary trading firms. In the RT market, the RT LMPs and the unit/resource dispatch schedules are calculated based on the updated supply offers and the actual operating conditions described by the state estimator. Proprietary trading firms can use virtual bids to arbitrage in the electricity market when the expected RT LMPs differ from the expected DA LMPs.

In this section, we develop a machine learning based trading strategy for proprietary trading firms to maximize their total profits with limited portfolio risks. It can also be used to evaluated the market efficiency based on the trading profitability across different markets.

5.1 Related Works

Market participants can buy or sell energy in the DA market with an explicit requirement to sell or buy it back in the RT market using virtual bids. Note that there is no requirement for such virtual bids to be backed by physical assets. There are two types of virtual bids, increment offer (INC) and decrement bid (DEC) [156]. INC (DEC) is a financial instrument that enables virtual bidders to sell (buy) energy in the DA market and buy (sell) the same amount of energy back in the RT market at the same pricing node [157].

Virtual transactions were introduced into the two-settlement electricity market to improve market efficiency, promote price convergence [22], provide hedging instruments [23], and enhance market liquidity. IPPs and LSEs could leverage DECs to hedge risks associated with generator forced outage, higher RT electric load and volatile RT LMPs. Proprietary trading firms use virtual bids to arbitrage in the electricity market when the expected RT LMPs differ from the expected DA LMPs.

To further drive price convergence in the electricity market, improve market efficiency and increase profitability, it is imperative for proprietary trading firms to design virtual bids portfolio trading strategies that maximize the expected earnings and minimize risks. Virtual traders in proprietary trading firms regularly place speculative virtual bids to

arbitrage the differences between DA and RT LMPs based on their knowledge of the electricity market and the forecasts for the key factors that influence electricity prices. In fact, the cleared virtual transactions in the five major electricity markets in the U.S. was 13% [156] of the total load. In this section, we aim to develop a machine learning framework for algorithmic trading with virtual bids. We are interested in exploring if the machine learning based approach could find profitable virtual trading strategies in electricity markets.

Only a few researchers have studied portfolio trading strategies with virtual transactions from the perspective of proprietary trading firms. In [158], the portfolio optimization problem with virtual bids is formulated as a multi-armed bandit problem and solved by the algorithm referred to as dynamic programming on discrete set (DPDS). It was shown that DPDS consistently outperforms benchmark heuristic methods [159] when only the historical LMPs are considered. A risk-constrained virtual bids portfolio trading strategy is developed to empirically test for the efficiency of the California Independent System Operator market [157]. The existence of a profitable trading strategy with virtual transactions is explored via hypothesis testing in [22].

5.2 Portfolio Optimization with Virtual Bids

5.2.1 Modeling of Virtual Bids

In this subsection, we will model two types of virtual transactions, incremental offers (INCs) and decremental bids (DECs) in detail.

Let $\lambda_{i,h}^{DA}$ and $\lambda_{i,h}^{RT}$ denote the DA LMP and the RT LMP for node i and hour h. Define $\lambda_{i,h}^{dif} = \lambda_{i,h}^{RT} - \lambda_{i,h}^{DA}$ as the price spread between the DA LMP and the RT LMP. Let

$\lambda_{i,h}^{bid,I}$ and $\lambda_{i,h}^{bid,D}$ represent the bid price (\$/MWh) for INCs and DECs for node i and hour h.

INCs are also called virtual supply offers, which sell energy in the DA market and must buy back the same amount of energy in the RT market [160]. A virtual supply offer will be cleared if the bid price is lower than the DA LMP. The expected earning of an INC $E[r_{i,h}^{I}]$ is

$$E[r_{i,h}^{I}] = E\big[-\lambda_{i,h}^{dif}\mathbb{1}(\lambda_{i,h}^{bid,I} \leq \lambda_{i,h}^{DA})\big] \tag{5.1}$$

DECs are also called virtual demand bids, which buy energy in the DA market and must sell the same amount of energy in the RT market as a price-taker. A virtual demand bid will be cleared if the bid price is higher than the DA LMP. The expected earning of a DEC $E[r_{i,h}^{D}]$ is

$$E[r_{i,h}^{D}] = E\big[\lambda_{i,h}^{dif}\mathbb{1}(\lambda_{i,h}^{bid,D} \geq \lambda_{i,h}^{DA})\big] \tag{5.2}$$

5.2.2 Budget and Risk Constrained Portfolio Optimization

Before the DA market closes, each day a trader determines a portfolio of virtual bids to be submitted to the DA market which maximizes the expected portfolio earnings subject to budget and risk constraints as follows:

$$\max_{\boldsymbol{z}} \sum_{i=1}^{N}\sum_{h=1}^{24} E[r_{i,h}^{I} z_{i,h}^{I} + r_{i,h}^{D} z_{i,h}^{D}] \tag{5.3}$$

s.t.

$$\sum_{i=1}^{N}\sum_{h=1}^{24} [Prox_{(i,h)}^{I} z_{i,h}^{I} + Prox_{(i,h)}^{D} z_{i,h}^{D}] \leq B \tag{5.4}$$

$$\sum_{h=1}^{24} CVaR_{\alpha}(f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}})) \leq C \tag{5.5}$$

$$f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}}) = -\sum_{i=1}^{N} [r_{i,h}^I z_{i,h}^I + r_{i,h}^D z_{i,h}^D] \tag{5.6}$$

$$z_{i,h}^I, z_{i,h}^D \in \{0, 1\} \tag{5.7}$$

$z_{i,h}^I$ and $z_{i,h}^D$ are the binary variables indicating whether or not the corresponding virtual bids with a quantity of 1MWh are selected for submission. The portfolio loss function f_h can be calculated as the summation of the loss of individual virtual bids (5.6), where $\boldsymbol{z_h}$ is the binary decision vector for the submission of virtual bids for all nodes at hour h and $\boldsymbol{\lambda_h^{dif}}$ is the vector of random price spreads for all nodes at hour h. Equation (5.4) is the budget constraint, where $Prox_{(i,h)}^I$ and $Prox_{(i,h)}^D$ denotes the DA virtual bid financial assurance proxy for INC and DEC at node i and hour h respectively, and B is the total portfolio budget limit. The virtual proxy is utilized by ISO for calculating the financial assurance requirements for virtual transactions [161]. Equation (5.5) enforces the portfolio risk constraint, where the risk metric is selected as the conditional value-at-risk (CVaR) with confidence level α [162]. The details about estimating the expected earnings of virtual transactions and risk-constrained optimization are described in Section 5.3 and 5.4 respectively.

5.3 Estimation of Expected Earnings of Virtual Transactions

5.3.1 Benchmark Algorithm: An Online Learning Approach

The Online Learning (OL) approach proposed in [158] and [159] is used as the benchmark algorithm in this section. The OL approach assumes that traders only have

access to historical DA LMPs, RT LMPs, and bid prices when determining the optimal virtual bids. In addition, the payoff of a virtual bid with a given bidding price has the same distribution as that of the historical bids. To make the OL approach consistent with the portfolio optimization framework presented in (5.3)-(5.7), we add the portfolio risk constraint and modify the portfolio budget constraint. Note that most of the ISOs in the U.S. use the virtual proxy for the budget constraint as in equation (5.4) rather than the bid price as shown in [158].

In this subsection, we follow the convention of notations in [158] and [159] to make the payoff function for INCs and DECs consistent. The DA, RT, and bid prices of INCs are transformed as $\overline{\lambda} - \lambda_{i,h}^{DA}$, $\overline{\lambda} - \lambda_{i,h}^{RT}$, and $\overline{\lambda} - \lambda_{i,h}^{bid,I}$ respectively, where $\overline{\lambda}$ is the upper bound of DA LMP. Similarly, the DA, RT and bid prices for DECs are transformed as $\lambda_{i,h}^{DA} - \underline{\lambda}$, $\lambda_{i,h}^{RT} - \underline{\lambda}$, and $\lambda_{i,h}^{bid,D} - \underline{\lambda}$ respectively, where $\underline{\lambda}$ is the lower bound of DA LMP. This way, the functional form of payoffs for the INCs is the same as that for the DECs.

In the OL approach, the expected payoff of a virtual bid with a given bid price is calculated as the average empirical payoff based on the historical DA and RT LMPs. The expected payoff is calculated sequentially and adaptively based on the new market information. The virtual transaction at each node and each hour of a day is treated as a single product. Let $\boldsymbol{\lambda}_t^{(k)}$ denote the bidding price vector for the kth product from day 1 to day t with the bidding prices sorted in ascending order. Let $\boldsymbol{r}_t^{(k)}$ denote the vector of expected payoff for a virtual transaction corresponding to the bid price in $\boldsymbol{\lambda}_t^{(k)}$.

After receiving the new information about bidding price on day t for product k, $\boldsymbol{\lambda}_t^{(k)}$ can be updated based on $\boldsymbol{\lambda}_{t-1}^{(k)}$ as:

$$\boldsymbol{\lambda}_t^{(k)} = [\boldsymbol{\lambda}_{t-1}^{(k)}(1:i_k), \lambda_{t,k}, \boldsymbol{\lambda}_{t-1}^{(k)}(i_k+1:t-1)] \tag{5.8}$$

where $\lambda_{t,k}$, is inserted at the i_k+1th index with $i_k = \max_i \boldsymbol{\lambda}_{t-1}^{(k)}(i) < \lambda_{t,k}$. The symbol $(\cdot)$ denotes the operation of fetching elements by indexes.

The vector for corresponding expected payoff $\boldsymbol{r}_t^{(k)}$ can be updated based on $\boldsymbol{r}_{t-1}^{(k)}$ as:

$$\boldsymbol{r}_t^{(k)} = [\frac{t-1}{t}\boldsymbol{r}_{t-1}^{(k)}(1:i_k), \frac{t-1}{t}\boldsymbol{r}_{t-1}(i_k:t-1) + \frac{1}{t}(\lambda_{t,k}^{RT} - \lambda_{t,k}^{DA})] \tag{5.9}$$

where $\lambda_{t,k}^{RT}$ and $\lambda_{t,k}^{DA}$ are the DA LMP and the RT LMP for product k at day t respectively.

For a trader following the OL approach, the bid price needs to be selected for each product k. Obviously, the bid price with the highest expected payoff will be selected. The expected payoff of bidding product k can then be updated as:

$$E[r^k] = max(\boldsymbol{r}_t^{(k)}) \tag{5.10}$$

5.3.2 A Machine Learning Framework

Instead of relying only on historical DA LMPs, RT LMPs, and bid prices to estimate the expected price spreads and the payoffs of virtual transactions, the key features that influence the formation of DA and RT prices should also be included in the prediction model. For example, load forecast, generation capacity, transmission outages, fuel prices, and meteorological forecast are all crucial in determining the distribution of price spreads and payoffs of virtual transactions. A machine learning framework based on MDN is proposed in this subsection to infer the distribution of the price spreads and the expected payoffs for virtual bids based on these key features.

Note that under the machine learning framework for algorithmic trading, the virtual bids are treated as self-schedules. In other words, the bid prices of INCs are selected to be the price floor of the DA LMP and the bid prices for DECs are selected to be the price cap of the DA LMP.

A. Input Features

The DA and RT LMPs are determined through the market clearing processes which involve solving security-constrained unit commitment problems and security-constrained economic dispatch problems in the DA and RT markets. The key features that influence the price spreads between DA and RT LMPs can be categorized into three groups. The first group includes all the meteorological variables such as system-wide/zonal temperature, dew point, cloud cover, and wind speed. The second group includes all the relevant fuel prices for natural gas, coal, and diesel. The third group includes the system variables such as the forecast for system/zonal demand, the available generation capacity by fuel type, and transmission outages. Note that only the forecast for these variables will be used for choosing the virtual bids to be submitted to the DA market.

B. Modeling Price Spread with Mixture Density Networks

The price spreads between DA and RT LMPs have extremely high volatility and spikiness [163]. For illustrative purposes, the histogram of the price spreads on a sample node in ISO New England (ISO-NE) is shown in Fig. 5.1. The frequency of occurrence for price spikes is not visible on the histogram. The price spikes are labeled by the red circles.

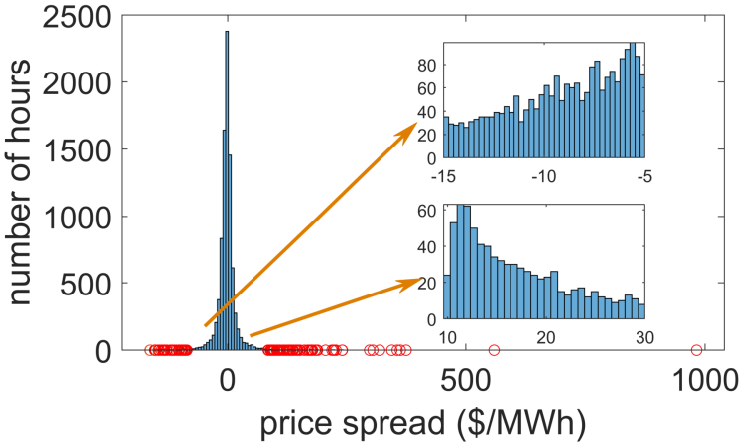

Figure 5.1: Histogram of price spreads of a sample node in year 2015

The two zoomed-in subplots for price ranges $(10, 30)$ and $(-15, -5)$ show that the price spread has a multimodal distribution. The typical neural network model assumes that the output variable has a Gaussian distribution with a mean dependent on the input variables. This Gaussian assumption can lead to very poor price spread predictions.

To deal with multimodality, we adopt mixture density network (MDN) to model the conditional probability distributions of the price spreads. In MDN, a Gaussian mixture distribution is assumed for the conditional distribution of the price spread as follows:

$$p(y|X) = \sum_{c=1}^{N_c} \pi_c(X)\mathcal{N}(y|\mu_c(X), \sigma_c^2(X)) \tag{5.11}$$

where X and y denote the input features and the prediction target. N_c is the total number of components. $\pi_c(X)$, $\mu_c(X)$, and $\sigma_c(X)$ are the feature dependent weight, mean value and standard deviation for component c respectively. The loss function of MDN is the negative logarithm of the likelihood function:

$$L(W) = -\sum_{n=1}^{N_s} \ln \sum_{c=1}^{N_c} \pi_c(x_n, W)\mathcal{N}(y_n|\mu_c(x_n, W), \sigma_c^2(x_n, W)) \tag{5.12}$$

where N_s is the total number of data samples. x_n and y_n are the features and the price prediction targets of sample n, while W denotes the network parameters. In order to make the summation of weights equal to 1, the softmax activation function is applied to the output layers for π_c. To make the estimate for standard deviation positive, the exponential activation function is adopted for the output layer of σ_c.

C. Data Preprocessing

Data preprocessing is crucial to minimize noise, capture nonlinear relationships, and flatten the distribution of variables which makes the learning process of neural networks more efficient and robust. Given that the spreads between the DA and RT LMPs are extremely volatile and spiky, the sigmoid function $f(x) = 1/(1 + exp(-x/\theta))$ is leveraged to scale the output price spread data and flatten the price spread distribution. Batch normalization is applied to the inputs.

5.4 Risk-Constrained Portfolio Optimization

The risk-constrained portfolio optimization problem (5.3)-(5.7) formulated in Section 5.2 is a mixed integer convex optimization problem. This is because the our chosen coherent risk measure [164], CVaR of the portfolio, is a convex function with respect to the positions in virtual bids. In this section, we will convert the mixed integer convex optimization problem into a mixed integer linear programming (MILP) problem.

First, we briefly review the definition of two risk metrics. The probability of the portfolio loss $f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}})$ at hour h not exceeding ζ_h is defined as

$$\Psi(\boldsymbol{z_h}, \zeta_h) = \int_{f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}}) \leq \zeta_h} p(\boldsymbol{\lambda_h^{dif}}) d\boldsymbol{\lambda_h^{dif}} \tag{5.13}$$

The value at risk (VaR), i.e., the probability of loss not exceeding ζ_h with confidence level α is defined as

$$VaR_\alpha(\boldsymbol{z_h}) = \min\{\zeta_h : \Psi(\boldsymbol{z_h}, \zeta_h) \geq \alpha\} \tag{5.14}$$

The $CVaR_\alpha$ of the portfolio loss can then be defined as

$$CVaR_\alpha(f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}})) = \frac{1}{1-\alpha} \int_{f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}}) \geq VaR_\alpha(\boldsymbol{z_h})} f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}}) p(\boldsymbol{\lambda_h^{dif}}) d\boldsymbol{\lambda_h^{dif}} \tag{5.15}$$

It has been proven that function $F_\alpha(\boldsymbol{z_h}, \zeta_h)$ is an upper bound of $CVaR_\alpha$ [165].

$$F_\alpha(\boldsymbol{z_h}, \zeta_h) = \frac{1}{1-\alpha} \int_{\lambda_h} [f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}}) - \zeta_h]^+ p(\boldsymbol{\lambda_h^{dif}}) d\boldsymbol{\lambda_h^{dif}} \tag{5.16}$$

By using the historical LMP samples, $F_\alpha(\boldsymbol{z_h}, \zeta_h)$ can be further simplified [166] as

$$F_\alpha(\boldsymbol{z_h}, \zeta_h) = \zeta_h + \frac{1}{(1-\alpha)N_s} \sum_{j=1}^{N_s} [f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_{h,j}^{dif}}) - \zeta_h]^+ \tag{5.17}$$

where N_s is the total number of samples. By introducing dummy variable u_h^j for sample j at hour h, equation (5.17) can be transformed as:

$$F_\alpha(\boldsymbol{u_h}, \zeta_h) = \zeta_h + \frac{1}{(1-\alpha)N_s} \sum_{j=1}^{N_s} u_h^j \tag{5.18}$$

$$u_h^j \geq f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_{h,j}^{dif}}) - \zeta_h \tag{5.19}$$

$$u_h^j \geq 0 \tag{5.20}$$

Hence, it can be shown [165] that CVaR of the portfolio loss can be determined from the formula below:

$$CVaR_\alpha(f_h(\boldsymbol{z_h}, \boldsymbol{\lambda_h^{dif}})) = \min_{\zeta_h} F_\alpha(\boldsymbol{z_h}, \zeta_h) \tag{5.21}$$

Therefore, the original risk-constrained virtual transactions portfolio optimization problem can be reformulated as

$$\max_{\boldsymbol{z},\boldsymbol{\zeta},\boldsymbol{u}} \sum_{i=1}^{N} \sum_{h=1}^{24} E[r_{i,h}^{I} z_{i,h}^{I} + r_{i,h}^{D} z_{i,h}^{D}] \tag{5.22}$$

s.t.

$$\sum_{h=1}^{24} F_{\alpha}(\boldsymbol{u_h}, \zeta_h) \leq C \tag{5.23}$$

$$(5.4), (5.6), (5.7), (5.18) - (5.20)$$

The optimization problem is now a MILP which can be tackled by commercial solvers such as Gurobi and CPLEX.

5.5 Numerical Study

The performances of the proposed virtual transaction bidding strategy based on MDN and the benchmark OL approach are evaluated with the electricity market managed by ISO-NE. The historical LMPs and input variables such as oil and gas prices forecasts, total system demand forecast, wind generation forecast, and weather forecasts, are taken from online data repositories [167, 168]. The DA and RT LMPs from the 994 pricing nodes in ISO-NE are collected. Three years of historical data from year 2015 to 2017 are gathered. The first year's data is used for training and validation purpose. The rest of the data are used for out-of-sample testing. In the testing process, the proposed MDN model is updated on a weekly basis. The costs associated with virtual bids in ISO-NE include transaction fee and net commitment period compensation (NCPC). NCPC is a payment to generators, dispatchable-asset-related demands, demand response resources or the external transactions that did not recover their effective offer costs from the energy market. The transaction fee

for virtual bids is \$0.065/MWh. The NCPCs are \$1.25/MWh in 2016 and \$0.77/MWh in 2017 for ISO-NE.

5.5.1 Profitability of Algorithmic Trading Strategies

Both the OL approach and the proposed MDN based trading strategy are used to select portfolios of virtual transactions in ISO-NE on a daily basis between January 1, 2016 and December 31, 2017. Two sets of daily portfolio budget and risk constraints of \$50k and \$100k are tested. The confidence level α of CVaR is set as 0.95. The gain, the trading costs, and the net gains are reported in Table 5.1. It can be seen from the table that our proposed MDN based trading strategy consistently outperforms the OL approach in both years and under both sets of budget and risk constraints. When transaction costs and NCPC are taken into consideration, the net gains of our proposed trading strategy are \$1.17 million and \$2.38 million in 2016 and 2017 with a \$100k risk and budget limit. The trading performance results suggest that the MDN provides more accurate prediction for price spreads than historical averages calculated by the OL approach. Finally, note that when the daily budget and risk limit increases, the net gains of the proposed MDN based trading strategy also increases.

5.5.2 Influence of Rare Events and Changes in Market Rules

The cumulative net gains of our proposed and benchmark virtual bids trading strategies with \$100k daily budget and risk limit for year 2016 and 2017 are depicted in Figure 5.2. As shown in the figure, the performance of the proposed and benchmark trading

Table 5.1: Performance comparison of algorithmic trading strategies

Strategy	Year	Budget / Risk ($)	Gain (million $)	Fees & Uplift (million $)	Net Gain (million $)
OA	2016	50k / 50k	1.3375	1.2211	0.1164
		100k / 100k	1.8406	2.2876	−0.4469
	2017	50k / 50k	1.0506	0.7149	0.3356
		100k / 100k	1.6967	1.3496	0.3472
MDN	2016	50k / 50k	2.2359	1.4087	0.8271
		100k / 100k	3.9504	2.7804	1.1701
	2017	50k / 50k	2.4221	0.8806	1.5415
		100k / 100k	4.1088	1.7319	2.3770

strategy can be underwhelming during certain events/periods. It indicates that rare market events and changes in market operation rules not captured by the machine learning model have a significant impact on the performance of the trading algorithms.

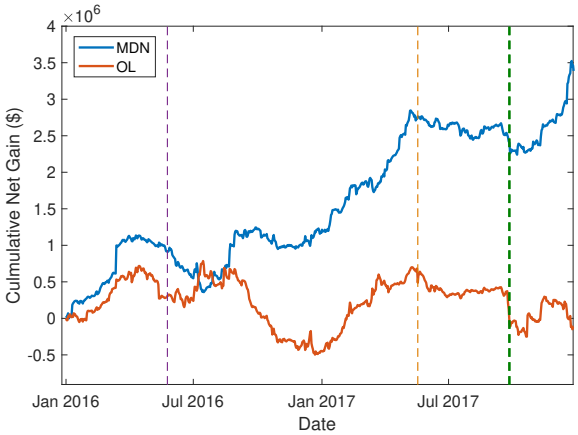

Figure 5.2: Cumulative net gains of the proposed and the benchmark trading strategies from 2016 to 2017 with $100k daily budget and risk limit

The first slump of MDN algorithm in trading performance is due to the implementation of Do-Not-Exceed (DNE) dispatch rules by ISO-NE on May 25, 2016 (labeled

by the purple dash line). Before the change in market rules, the practice of manual curtailment for renewable generation resource was adopted by system operators. Under the DNE dispatch rules, renewable generation units must submit supply offers into the DA market. These offers are allowed to settle the DA LMPs on the corresponding pricing nodes when congestion happens. It can be seen that the cumulative net gain of the proposed trading strategy takes a dip right after DNE's implementation. The sharp dip in net gain of the benchmark algorithm on May 18, 2017 (labeled by the orange dash line) and the big drops of MDN algorithm between September 24 and September 27, 2017 (labeled by the green dash lines) are caused by price spikes in RT. These price spikes result from forced outages in RT, which are difficult to predict on a DA basis.

5.6 Conclusion and Future Works

In this section, a data-driven algorithmic trading strategy is developed for virtual transactions in electricity markets. A budget and risk constrained portfolio optimization problem is formulated to select the virtual bids to be submitted. A mixture density network model is developed to forecast the price spreads between DA and RT LMPs. Backcasting results with ISO-NE's market data show that the proposed MDN based algorithmic trading strategy is much more profitable than the benchmark OL approach. In the future, we will consider the price sensitivity in the portfolio optimization framework and develop a machine learning framework to jointly model the price spreads of all pricing nodes. Furthermore, a comprehensive comparison of the profitability across different electricity markets will be performed to evaluate their relative efficiency.

Chapter 6

Conclusions

In this dissertation, we investigate the optimization and control problems in smart grids for different roles. For system operators, we develop the three-phase optimal power flow algorithms to coordinate the distributed energy resources (DERs). Furthermore, we design the deep reinforcement learning (DRL) algorithms for Volt-VAR control in distribution systems. For data centers, we present the proactive demand participation schemes to provide the phase balancing service at a distribution system level and the frequency regulation service at a transmission level. For proprietary trading firms, we develop a machine learning based framework to maximize their total profits with limited portfolio risks. In conclusion, the major work of this dissertation is summarized as follows:

In Chapter 2, we develop a three-phase iterative direct current optimal power flow (DCOPF) algorithm with fictitious nodal demand. The definition of locational marginal prices (LMPs) is extended to three-phase distribution systems. The proposed DCOPF algorithm is effective in coordinating the operations of DERs and managing phase imbalance

and thermal overloading. Furthermore, an innovative three-phase alternating current optimal power flow (ACOPF) algorithm is developed by synergistically combining the convex iteration technique and the chordal based conversion algorithm. We also propose a greedy algorithm to find an appropriate grid partitioning scheme which results in lower computational complexity. The computational efficiency and scalability of the proposed algorithm and the optimality of the solutions are validated on the IEEE test feeders.

In Chapter 3, the phase balancing problem in distribution networks is tackled by shifting computational loads among the servers connected to three different phase wires in a data center. An iterative scheme is developed to coordinate the operations of data centers and DERs within a DSO managed electricity market. The sensitivities of three-phase LMPs are derived and embedded into the scheme as damp factors. The coordination strategy for data centers and DERs is very effective in reducing phase unbalance, improving distribution network operational efficiency and reliability. Secondly, a comprehensive framework for a data center to provide frequency regulation services is proposed, which includes an hour-ahead bidding strategy and a real-time signal following algorithm. The proposed realistic bi-linear server power consumption model and rule-based data center power consumption control algorithm not only enable accurate frequency regulation signal following but also limit the degradation in QoS. In addition, the dummy computing load is introduced to increase the amount of frequency regulation service provision and final compensation.

In Chapter 4, the DRL approaches are adopted to learn a Volt-VAR control (VVC) policy, which minimizes the total operation costs while satisfying the physical operation constraints. The VVC problem is formulated as a constrained Markov decision process

and solved by the on-policy constrained policy optimization at first. To further improve the sample efficiency, an off-policy algorithm is proposed by synergistically combining the merits of the method of multipliers and soft actor-critic algorithm. By designing the policy neural network with a device-decoupled structure, the number of parameters in our proposed method increases linearly with the number of voltage regulating devices. In contrast to physical model-based VVC algorithms, our proposed DRL approaches are model-free and does not rely on complete and accurate distribution network topology models or parameters. In addition, the numerical study results show that the policy gradient methods are capable of learning near-optimal solutions and determining control actions much faster than the optimization-based approaches.

In Chapter 5, the research frontier of algorithmic trading with virtual bids is further pushed. First, instead of relying only on historical LMPs to model and estimate the DA and RT price spreads and payoff of virtual bids, a mixture density network is developed to infer the conditional distribution of nodal price spreads given the fundamental inputs such as the forecasts of demand, fuel prices and weather. Second, a risk-constrained portfolio optimization problem for virtual bids is formulated and efficiently solved with a finite number of scenarios. The machine learning framework for algorithmic trading with virtual bids is established by synergistically combining the risk-constrained portfolio optimization framework and the mixture density network model. The proposed trading strategy also lays the fountain for the future research of market efficiency evaluation.

Bibliography

[1] EIA, "Form EIA-861M detailed data," [Online] https://www.eia.gov/electricity/data/eia861m/.

[2] ——, "Most utility-scale batteries in the United States are made of lithium-ion," [Online] https://www.eia.gov/todayinenergy/detail.php?id=41813.

[3] Statista, "Plug-in electric vehicle sales in the U.S. from 2014 to 2018," [Online] https://www.statista.com/statistics/416672/united-states-projected-plug-in-electric-vehicles-sales/.

[4] S. R. Group, "Hyperscale data center count passed the 500 milestone in Q3," [Online] https://www.srgresearch.com/articles/hyperscale-data-center-count-passed-500-milestone-q3.

[5] A. Shehabi, S. J. Smith, D. A. Sartor, R. E. Brown, M. Herrlin, J. G. Koomey, E. R. Masanet, N. Horner, I. L. Azevedo, and W. Lintner, "United States data center energy usage report," Lawrence Berkeley National Laboratory, Tech. Rep., Jun. 2016.

[6] J. Koomey, P. Turner, J. Stanley, and B. Taylor, "A simple model for determining true total cost of ownership for data centers," White Paper, Uptime Institute, Jan. 2007.

[7] EIA, "How many smart meters are installed in the United States, and who has them?" [Online] https://www.eia.gov/tools/faqs/faq.php?id=108&t=3.

[8] B. Moradzadeh and K. Tomsovic, "Two-stage residential energy management considering network operational constraints," *IEEE Trans. Smart Grid*, vol. 4, no. 4, pp. 2339–2346, Dec. 2013.

[9] T. Wei, Q. Zhu, and N. Yu, "Proactive demand participation of smart buildings in smart grid," *IEEE Trans. Comput.*, vol. 65, no. 5, pp. 1392–1406, May 2016.

[10] A. von Jouanne and B. Banerjee, "Assessment of voltage unbalance," *IEEE Trans. Power Del.*, vol. 16, no. 4, pp. 782–790, Oct. 2001.

[11] M. W. Davis, R. Broadwater, and J. Hambrick, "Modeling and testing of unbalanced loading and voltage regulation," NREL, Tech. Rep., Jul. 2007.

[12] California ISO, "Q3 2017 report on market issues and performance," Dec. 2017.

[13] R. Xu, R. Melhem, and D. Mossé, "A unified practical approach to stochastic DVS scheduling," in *Proceedings of the 7th ACM International Conference on Embedded Software*, 2007, pp. 37–46.

[14] S. Herbert and D. Marculescu, "Analysis of dynamic voltage/frequency scaling in chip-multiprocessors," in *Proceedings of the 2007 International Symposium on Low Power Electronics and Design*, 2007, pp. 38–43.

[15] K. Li, "Power and performance management for parallel computations in clouds and data centers," *Journal of Computer and System Sciences*, vol. 82, no. 2, pp. 174 – 190, Mar. 2016.

[16] A. H. Mahmud, Y. He, and S. Ren, "BATS: budget-constrained autoscaling for cloud performance optimization," in *The 2014 ACM International Conference on Measurement and Modeling of Computer Systems*, 2014, pp. 563–564.

[17] A. Verma, P. Ahuja, and A. Neogi, "Pmapper: power and migration cost aware application placement in virtualized systems," in *Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware*, Nov. 2008, pp. 243–264.

[18] M. Lin, Z. Liu, A. Wierman, and L. L. H. Andrew, "Online algorithms for geographical load balancing," in *2012 International Green Computing Conference (IGCC)*, Jun. 2012, pp. 1–10.

[19] W. Wang, N. Yu, B. Foggo, J. Davis, and J. Li, "Phase identification in electric power distribution systems by clustering of smart meter data," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2016, pp. 259–265.

[20] B. Foggo and N. Yu, "A comprehensive evaluation of supervised machine learning for the phase identification problem," *World Acad. Sci. Eng. Technol. Int. J. Comput. Syst. Eng*, vol. 12, no. 6, 2018.

[21] W. Wang, N. Yu, and Z. Lu, "Advanced metering infrastructure data driven phase identification in smart grid," *GREEN 2017 Forward*, pp. 16–23, 2017.

[22] F. A. W. Akshaya Jha, "Testing for market efficiency with transactions costs: An application to convergence bidding in wholesale electricity markets," *Industrial Organization Seminar, Yale University*, May 2013.

[23] W. W. Hogan, "Virtual bidding and electricity market design," *The Electricity Journal*, vol. 29, pp. 33–47, Jun. 2016.

[24] W. Wang and N. Yu, "LMP decomposition with three-phase DCOPF for distribution system," in *2016 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia)*, Nov. 2016, pp. 1–8.

[25] ——, "Chordal conversion based convex iteration algorithm for three-phase optimal power flow problems," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1603–1613, Mar. 2018.

[26] ——, "Phase balancing in power distribution network with data center," *SIGMETRICS Perform. Eval. Rev.*, vol. 45, no. 2, pp. 64–69, Oct. 2017.

[27] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *ICML*, vol. 70, Aug. 2017, pp. 22–31.

[28] W. Wang, N. Yu, J. Shi, and Y. Gao, "Volt-VAR control in power distribution systems with deep reinforcement learning," in *IEEE SmartGridComm*, Oct. 2019, pp. 1–7.

[29] W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe off-policy deep reinforcement learning algorithm for Volt-VAR control in power distribution systems," *IEEE Trans. Smart Grid*, pp. 1–11, Dec. 2019.

[30] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *ICML*, vol. 80, Stockholmsmässan, Stockholm Sweden, Jul. 2018, pp. 1861–1870.

[31] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.

[32] H. v. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *AAAI*, Feb. 2016, pp. 2094–2100.

[33] W. Wang and N. Yu, "A machine learning framework for algorithmic trading with virtual bids in electricity markets," in *IEEE PES GM*, Aug. 2019, pp. 1–8.

[34] S. Baltaoglu, L. Tong, and Q. Zhao, "Online learning of optimal bidding strategy in repeated multi-commodity auctions," in *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017, vol. 30, pp. 4507–4517.

[35] W. Wang, A. Abdolrashidi, N. Yu, and D. Wong, "Frequency regulation service provision in data center with computational flexibility," *Applied Energy*, vol. 251, p. 113304, 2019.

[36] P. M. Sotkiewicz and J. M. Vignolo, "Nodal pricing for distribution networks: efficient pricing for efficiency enhancing DG," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 1013–1014, May 2006.

[37] A. J. Conejo, J. M. Morales, and L. Baringo, "Real-time demand response model," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 236–242, Dec. 2010.

[38] N. Yu, T. Wei, and Q. Zhu, "From passive demand response to proactive demand participation," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Aug. 2015, pp. 1300–1306.

[39] M. Hong, X. Yu, N. P. Yu, and K. A. Loparo, "An energy scheduling algorithm supporting power quality management in commercial building microgrids," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 1044–1056, March 2016.

[40] H. W. Dommel and W. F. Tinney, "Optimal power flow solutions," *IEEE Trans. Power App. Syst.*, vol. PAS-87, no. 10, pp. 1866–1876, Oct. 1968.

[41] O. Alsac and B. Stott, "Optimal load flow with steady-state security," *IEEE Trans. Power App. Syst.*, vol. PAS-93, no. 3, pp. 745–751, May 1974.

[42] G. C. Contaxis, C. Delkis, and G. Korres, "Decoupled optimal load flow using linear and quadratic programming," *IEEE Trans. Power Syst.*, vol. 1, no. 2, pp. 1–7, May 1986.

[43] B. Ghaddar, J. Marecek, and M. Mevissen, "Optimal power flow as a polynomial optimization problem," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 539–546, Jan. 2016.

[44] I. M. Nejdawi, K. A. Clements, and P. W. Davis, "An efficient interior-point method for sequential quadratic programming based optimal power flow," *IEEE Trans. Power Syst.*, vol. 15, no. 4, pp. 1179–1183, Nov. 2000.

[45] P. E. O. Yumbla, J. M. Ramirez, and C. A. C. Coello, "Optimal power flow subject to security constraints solved with a particle swarm optimizer," *IEEE Trans. Power Syst.*, vol. 23, no. 1, pp. 33–40, Feb. 2008.

[46] X. Bai, H.Wei, K. Fujisawa, and Y. Wang, "Semidefinite programming for optimal power flow problems," *Int. J. Elect. Power Energy Syst.*, vol. 30, no. 6, pp. 383–392, Sep. 2008.

[47] J. Lavaei, D. Tse, and B. Zhang, "Geometry of power flows in tree networks," in *Proc. 2012 IEEE Power & Energy Soc. General Meeting*, Jul. 2012, pp. 1–8.

[48] J. Lavaei and S. H. Low, "Zero duality gap in optimal power flow problem," *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 92–107, Feb. 2012.

[49] S. H. Low, "Convex relaxation of optimal power flow-part II: Exactness," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 2, pp. 177–189, Jun. 2014.

[50] L. Gan, N. Li, U. Topcu, and S. H. Low, "Exact convex relaxation of optimal power flow in radial networks," *IEEE Trans. Autom. Control*, vol. 60, no. 1, pp. 72–87, Jan. 2015.

[51] S. You and Q. Peng, "A non-convex alternating direction method of multipliers heuristic for optimal power flow," in *Smart Grid Communications, 2014 IEEE Int. Conf. on*, Nov. 2014, pp. 788–793.

[52] D. K. Molzahn, C. Josz, I. A. Hiskens, and P. Panciatici, "A Laplacian-based approach for finding near globally optimal solutions to OPF problems," *IEEE Trans. Power Syst.*, vol. PP, no. 99, pp. 1–1, Apr. 2016.

[53] R. Madani, S. Sojoudi, and J. Lavaei, "Convex relaxation for optimal power flow problem: Mesh networks," *IEEE Trans. Power Syst.*, vol. 30, no. 1, pp. 199–211, Jan. 2015.

[54] S. Bruno, S. Lamonaca, G. Rotondo, U. Stecchi, and M. L. Scala, "Unbalanced three-phase optimal power flow for smart grids," *IEEE Trans. Ind. Electron.*, vol. 58, no. 10, pp. 4504–4513, Oct. 2011.

[55] E. Dall'Anese, H. Zhu, and G. B. Giannakis, "Distributed optimal power flow for smart microgrids," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1464–1475, Sep. 2013.

[56] F. Li and R. Bo, "DCOPF-based LMP simulation: Algorithm, comparison with ACOPF, and sensitivity," *IEEE Trans. Power Syst.*, vol. 22, no. 4, pp. 1475–1485, Nov. 2007.

[57] J. Arrillaga and B. J. Harker, "Fast-decoupled three-phase load flow," *Electrical Engineers, Proceedings of the Institution of*, vol. 125, no. 8, pp. 734–740, Aug. 1978.

[58] R. A. Jabr, "Exploiting sparsity in SDP relaxations of the OPF problem," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 1138–1139, May 2012.

[59] D. K. Molzahn, J. T. Holzer, B. C. Lesieutre, and C. L. DeMarco, "Implementation of a large-scale optimal power flow solver based on semidefinite programming," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3987–3998, Nov. 2013.

[60] J. Dattorro, *Convex Optimization and Euclidean Distance Geometry.* CA: Meboo Publishing, 2005.

[61] R. Grone, C. R. Johnson, E. M., and H. Wolkowicz, "Positive definite completions of partial Hermitian matrices," *Linear Algebra and its Applications*, vol. 58, pp. 109–124, 1984.

[62] M. S. Andersen, A. Hansson, and L. Vandenberghe, "Reduced-complexity semidefinite relaxations of optimal power flow problems," *IEEE Trans. Power Syst.*, vol. 29, no. 4, pp. 1855–1863, Jul. 2014.

[63] L. Vandenberghe and M. S. Andersen, "Chordal graphs and semidefinite optimization," *Foundations and Trends in Optimization*, vol. 1, no. 4, pp. 241–433, 2015.

[64] J. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimiz. Meth. Soft.*, vol. 11, no. 1, pp. 625–653, 1999.

[65] ——, "Implementation of interior-point methods for mixed semidefinite and second order cone optimization problems," *Optimiz. Meth. Soft.*, vol. 17, no. 6, pp. 1105–1154, 2002.

[66] F. Alizadeh, J. Haeberly, and M. Overton, "A new primal-dual interior-point method for semidefinite programming," in *Proc. 5th SIAM Conf. on Applied Linear Algebra*, Philadelphia, 1994, pp. 113–117.

[67] Y. Nesterov and M. Todd, "Self-scaled barriers and interior-point methods for convex programming," *Mathematics of Operations Research*, vol. 22, no. 1, pp. 1–42, 1997.

[68] C. Helmberg, F. Rendl, R. Vanderbei, and H. Wolkowicz, "An interior-point method for semidefinite programming," *SIAM J. Optimiz.*, vol. 6, pp. 342–361, 1996.

[69] J. Sturm, "Primal-dual interior-point approach to semidefinite programming," Ph.D. dissertation, Tinbergen Institute of the Erasmus University, Rotterdam, The Netherlands, 1997.

[70] J. Lofberg, "YALMIP : a toolbox for modeling and optimization in MATLAB," in *CACSD*, Sep. 2004, pp. 284–289.

[71] A. M. Sasson, F. Viloria, and F. Aboytes, "Optimal load flow solution using the hessian matrix," *IEEE Trans. Power App. Syst.*, vol. PAS-92, no. 1, pp. 31–41, Jan. 1973.

[72] S. N. Talukdar, T. C. Giras, and V. K. Kalyan, "Decompositions for optimal power flows," *IEEE Trans. Power App. Syst.*, vol. PAS-102, no. 12, pp. 3877–3884, Dec. 1983.

[73] S. Granville, "Optimal reactive dispatch through interior point methods," *IEEE Trans. Power Syst.*, vol. 9, no. 1, pp. 136–146, Feb. 1994.

[74] H. Wei, H. Sasaki, J. Kubokawa, and R. Yokoyama, "An interior point nonlinear programming for optimal power flow problems with a novel data structure," *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 870–877, 1998.

[75] W. H. Kersting, "Radial distribution test feeders," in *2001 IEEE Power Engineering Society Winter Meeting.*, vol. 2, Jan 2001, pp. 908–912 vol.2.

[76] S. Herbert and D. Marculescu, "Analysis of dynamic voltage/frequency scaling in chip-multiprocessors," in *Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium on*, Aug. 2007, pp. 38–43.

[77] V. Spiliopoulos, S. Kaxiras, and G. Keramidas, "Green governors: a framework for continuously adaptive dvfs," in *2011 International Green Computing Conference and Workshops*, July 2011, pp. 1–8.

[78] D. Lo and C. Kozyrakis, "Dynamic management of turbomode in modern multi-core chips," in *2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)*, Feb. 2014, pp. 603–613.

[79] D. Meisner, B. T. Golden, and T. F. Wenisch, "Powernap: eliminating server idle power," in *Proceedings of the 14th International Conference on Architectural Support for Programming Languages and Operating Systems*, 2009, pp. 205–216.

[80] D. Meisner and T. F. Wenisch, "Dreamweaver: architectural support for deep sleep," in *Proceedings of the Seventeenth International Conference on Architectural Support for Programming Languages and Operating Systems*, New York, NY, USA, 2012, pp. 313–324.

[81] D. Wong, "Peak efficiency aware scheduling for highly energy proportional servers," in *Proceedings of the 43rd International Symposium on Computer Architecture*, 2016.

[82] M. Ghamkhari and H. Mohsenian-Rad, "Energy and performance management of green data centers: a profit maximization approach," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 1017–1025, Jun 2013.

[83] Y. Wang, X. Lin, and M. Pedram, "A stackelberg game-based optimization framework of the smart grid with distributed PV power generations and data centers," *IEEE Trans. Energy Convers.*, vol. 29, no. 4, pp. 978–987, Dec. 2014.

[84] N. H. Tran, D. H. Tran, S. Ren, Z. Han, E. N. Huh, and C. S. Hong, "How geo-distributed data centers do demand response: a game-theoretic approach," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 937–947, Mar. 2016.

[85] L. Yu, T. Jiang, Y. Zou, and Z. Sun, "Joint energy management strategy for geo-distributed data centers and electric vehicles in smart grid environment," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2378–2392, Sept. 2016.

[86] M. Ghamkhari and H. Mohsenian-Rad, "Data centers to offer ancillary services," in *IEEE International Conference on Smart Grid Communications*, Nov. 2012, pp. 436–441.

[87] Y. Shi, B. Xu, B. Zhang, and D. Wang, "Leveraging energy storage to optimize data center electricity cost in emerging power markets," in *ACM e-Energy*, 2016, pp. 18:1–18:13.

[88] S. Li, M. Brocanelli, W. Zhang, and X. Wang, "Data center power control for frequency regulation," in *IEEE Power Energy Society General Meeting*, Jul. 2013, pp. 1–5.

[89] H. Chen, A. K. Coskun, and M. C. Caramanis, "Real-time power control of data centers for providing regulation service," in *IEEE Conference on Decision and Control*, Dec. 2013, pp. 4314–4321.

[90] H. Chen, M. C. Caramanis, and A. K. Coskun, "Reducing the data center electricity costs through participation in smart grid programs," in *International Green Computing Conference*, Nov. 2014, pp. 1–10.

[91] ——, "The data center as a grid load stabilizer," in *Asia and South Pacific Design Automation Conference*, Jan. 2014, pp. 105–112.

[92] H. Chen, B. Zhang, M. C. Caramanis, and A. K. Coskun, "Data center optimal regulation service reserve provision with explicit modeling of quality of service dynamics," in *IEEE Conference on Decision and Control*, Dec. 2015, pp. 7207–7213.

[93] B. Aksanli and T. Rosing, "Providing regulation services and managing data center peak power budgets," in *Design, Automation Test in Europe Conference Exhibition*, Mar. 2014, pp. 1–4.

[94] I. Alaperä, S. Honkapuro, and J. Paananen, "Data centers as a source of dynamic flexibility in smart girds," *Applied Energy*, vol. 229, pp. 69–79, 2018.

[95] S. Li, M. Brocanelli, W. Zhang, and X. Wang, "Integrated power management of data centers and electric vehicles for energy and regulation market participation," *IEEE Trans. Smart Grid*, vol. 5, no. 5, pp. 2283–2294, 2014.

[96] M. Brocanelli, S. Li, X. Wang, and W. Zhang, "Joint management of data centers and electric vehicles for maximized regulation profits," in *2013 International Green Computing Conference Proceedings*, Jun. 2013, pp. 1–10.

[97] E. Castillo, A. J. Conejo, C. Castillo, R. Mínguez, and D. Ortigosa, "Perturbation approach to sensitivity analysis in mathematical programming," *Journal of Optimization Theory and Applications*, vol. 128, no. 1, pp. 49–74, 2006.

[98] A. J. Conejo, E. Castillo, R. Minguez, and F. Milano, "Locational marginal price sensitivities," *IEEE Trans. Power Syst.*, vol. 20, no. 4, pp. 2026–2033, Nov. 2005.

[99] W. H. Kersting, "Radial distribution test feeders," *IEEE Trans. on Power Syst.*, vol. 6, no. 3, pp. 975–985, 1991.

[100] N. Yu, C.-C. Liu, and J. Price, "Evaluation of market rules using a multi-agent system method," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 470–479, 2010.

[101] A. Bergen, *Power Systems Analysis*, ser. Prentice-Hall series in electrical and computer engineering. Prentice-Hall, 1986.

[102] PJM, "Energy & ancillary services market operations," [Online] http://www.pjm.com/~/media/documents/manuals/m11.ashx.

[103] X. Fan, W.-D. Weber, and L. A. Barroso, "Power Provisioning for a Warehouse-sized Computer," in *Proceedings of the 34th Annual International Symposium on Computer Architecture*, ser. ISCA '07, 2007, pp. 13–23.

[104] D. Wong and M. Annavaram, "Knightshift: Scaling the energy proportionality wall through server-level heterogeneity," in *Microarchitecture (MICRO), 2012 45th Annual IEEE/ACM International Symposium on*, 2012, pp. 119–130.

[105] ——, "Implications of high energy proportional servers on cluster-wide energy proportionality," in *2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)*, 2014.

[106] M. Ferdman, A. Adileh, O. Kocberber, S. Volos, M. Alisafaee, D. Jevdjic, C. Kaynak, A. D. Popescu, A. Ailamaki, and B. Falsafi, "Clearing the clouds: A study of emerging scale-out workloads on modern hardware," in *Proceedings of the 17th International*

Conference on Architectural Support for Programming Languages and Operating Systems, Mar. 2012, pp. 37–48.

[107] V. M. Weaver, M. Johnson, K. Kasichayanula, J. Ralph, P. Luszczek, D. Terpstra, and S. Moore, "Measuring energy and power with PAPI," in *Proceedings of the 41st International Conference on Parallel Processing Workshops*, Sept. 2012, pp. 262–268.

[108] D. Wong, "Peak efficiency aware scheduling for highly energy proportional servers," in *Proceedings of the 43th Annual International Symposium on Computer Architecture*, 2016.

[109] P. J. Brockwell, *Introduction to Time Series and Forecasting*. Springer-Verlag New York, 2002.

[110] Pierr Johnson, "With the public clouds of Amazon, Microsoft and Google, big data is the proverbial big deal," https://www.forbes.com/sites/johnsonpierr/2017/06/15/with-the-public-clouds-of-amazon-microsoft-and-google-big-data-is-the-proverbial-big-deal.

[111] G. Pierre, "Wiki data repository," [Online]. Available from: http://www.wikibench.eu/?page_id=60.

[112] Google, "Data center efficiency," https://www.google.com/about/datacenters/efficiency/internal/.

[113] Facebook, "Designing a very efficient data center," https://www.facebook.com/notes/facebook-engineering/designing-a-very-efficient-data-center/10150148003778920/.

[114] G. Urdaneta, G. Pierre, and M. van Steen, "Wikipedia workload analysis for decentralized hosting," *Computer Networks*, vol. 53, no. 11, pp. 1830–1845, Jul. 2009.

[115] R. Caruana, S. Lawrence, and L. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in *Proceedings of the 13th International Conference on Neural Information Processing Systems*, 2000, pp. 381–387.

[116] Y. Gao, J. Shi, W. Wang, and N. Yu, "Dynamic distribution network reconfiguration using reinforcement learning," in *IEEE SmartGridComm*, Oct. 2019, pp. 1–7.

[117] R. Diao, Z. Wang, D. Shi, Q. Chang, J. Duan, and X. Zhang, "Autonomous voltage control for grid operation using deep reinforcement learning," *arXiv preprint arXiv:1904.10597*, 2019.

[118] H. Ahmadi, J. R. Martí, and H. W. Dommel, "A framework for Volt-VAR optimization in distribution systems," *IEEE Trans. Smart Grid*, vol. 6, no. 3, pp. 1473–1483, May 2015.

[119] P. Li, H. Ji, C. Wang, J. Zhao, G. Song, F. Ding, and J. Wu, "Coordinated control method of voltage and reactive power for active distribution networks based on soft open point," *IEEE Trans. Sustain. Energy*, vol. 8, no. 4, pp. 1430–1442, Oct. 2017.

[120] M. H. K. Tushar and C. Assi, "Volt-VAR control through joint optimization of capacitor bank switching, renewable energy, and home appliances," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4077–4086, Sept. 2018.

[121] M. B. Liu, C. A. Canizares, and W. Huang, "Reactive power and voltage control in distribution systems with limited switching operations," *IEEE Trans. Power Syst.*, vol. 24, no. 2, pp. 889–899, May 2009.

[122] Y. Xu, Z. Y. Dong, R. Zhang, and D. J. Hill, "Multi-timescale coordinated voltage/VAR control of high renewable-penetrated distribution systems," *IEEE Trans. Power Syst.*, vol. 32, no. 6, pp. 4398–4408, Nov. 2017.

[123] W. Zheng, W. Wu, B. Zhang, and Y. Wang, "Robust reactive power optimisation and voltage control method for active distribution networks via dual time-scale coordination," *IET Generation, Transmission Distribution*, vol. 11, no. 6, pp. 1461–1471, May 2017.

[124] Z. Wang, J. Wang, B. Chen, M. M. Begovic, and Y. He, "MPC-based voltage/VAR optimization for distribution circuits with distributed generators and exponential load models," *IEEE Trans. Smart Grid*, vol. 5, no. 5, pp. 2412–2420, Sept. 2014.

[125] M. Falahi, K. Butler-Purry, and M. Ehsani, "Dynamic reactive power control of islanded microgrids," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3649–3657, Nov. 2013.

[126] K. E. Antoniadou-Plytaria, I. N. Kouveliotis-Lysikatos, P. S. Georgilakis, and N. D. Hatziargyriou, "Distributed and decentralized voltage control of smart distribution networks: Models, methods, and future research," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2999–3008, Nov. 2017.

[127] W. Lin, R. Thomas, and E. Bitar, "Real-time voltage regulation in distribution systems via decentralized PV inverter control," in *HICSS*, Jan. 2018.

[128] H. Zhu and H. J. Liu, "Fast local voltage control under limited reactive power: optimality and stability analysis," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3794–3803, Sept. 2016.

[129] H. Zhu and N. Li, "Asynchronous local voltage control in power distribution networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Mar. 2016, pp. 3461–3465.

[130] H. J. Liu, W. Shi, and H. Zhu, "Distributed voltage control in distribution networks: online and robust implementations," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 6106–6117, Nov. 2018.

[131] S. Bolognani, R. Carli, G. Cavraro, and S. Zampieri, "Distributed reactive power feedback control for voltage regulation and loss minimization," *IEEE Trans. Autom. Control*, vol. 60, no. 4, pp. 966–981, Apr. 2015.

[132] G. Cavraro and R. Carli, "Local and distributed voltage control algorithms in distribution networks," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1420–1430, Mar. 2018.

[133] B. Zhang, A. Y. S. Lam, A. D. Domínguez-García, and D. Tse, "An optimal and distributed method for voltage regulation in power distribution systems," *IEEE Trans. Power Syst.*, vol. 30, no. 4, pp. 1714–1726, Jul. 2015.

[134] A. Abessi, V. Vahidinasab, and M. S. Ghazizadeh, "Centralized support distributed voltage control by using end-users as reactive power support," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 178–188, Jan. 2016.

[135] L. Yu, D. Czarkowski, and F. de Leon, "Optimal distributed voltage regulation for secondary networks with DGs," *IEEE Trans. on Smart Grid*, vol. 3, no. 2, pp. 959–967, Jun. 2012.

[136] J. G. Vlachogiannis and N. D. Hatziargyriou, "Reinforcement learning for reactive power control," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1317–1325, Aug. 2004.

[137] Y. Xu, W. Zhang, W. Liu, and F. Ferrese, "Multiagent-based reinforcement learning for optimal reactive power dispatch," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 42, no. 6, pp. 1742–1751, Nov. 2012.

[138] H. Xu, A. D. Domínguez-García, and P. W. Sauer, "Optimal tap setting of voltage regulation transformers using batch reinforcement learning," *arXiv*, Jul. 2018.

[139] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[140] J. Schulman, S. Levine, P. Moritz, M. Jordan, and P. Abbeel, "Trust region policy optimization," in *ICML*, vol. 37, 2015, pp. 1889–1897.

[141] F. U. Nazir, B. C. Pal, and R. A. Jabr, "A two-stage chance constrained Volt/VAR control scheme for active distribution networks with nodal power uncertainties," *IEEE Trans. Power Syst.*, vol. 34, no. 1, pp. 314–325, Jan. 2019.

[142] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans. Power Delivery*, vol. 4, no. 2, pp. 1401–1407, Apr. 1989.

[143] J. W. Taylor and P. E. McSharry, "Short-term load forecasting methods: An evaluation based on european data," *IEEE Trans. Power Syst.*, vol. 22, no. 4, pp. 2213–2219, Nov. 2007.

[144] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv*, Jul. 2017. [Online]. Available: https://arxiv.org/abs/1707.06347

[145] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *ICML*, vol. 48, Jun. 2016, pp. 1928–1937.

[146] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv*, Sept. 2015. [Online]. Available: https://arxiv.org/abs/1509.02971

[147] T. Haarnoja, H. Tang, P. Abbeel, and S. Levine, "Reinforcement learning with deep energy-based policies," in *ICML*, vol. 70, Aug. 2017, pp. 1352–1361.

[148] B. D. Ziebart, "Modeling purposeful adaptive behavior with the principle of maximum causal entropy," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, PA, USA, 2010.

[149] O. Nachum, M. Norouzi, K. Xu, and D. Schuurmans, "Bridging the gap between value and policy based reinforcement learning," in *NIPS*, Feb. 2017, pp. 2775–2785.

[150] S. Fujimoto, H. van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *ICML*, vol. 80, Stockholmsmässan, Stockholm Sweden, Jul. 2018, pp. 1587–1596.

[151] C. Tessler, D. J. Mankowitz, and S. Mannor, "Reward constrained policy optimization," *arXiv*, May 2018.

[152] V. Borkar, "An actor-critic algorithm for constrained Markov decision processes," *Systems and Control Letters*, vol. 54, no. 3, pp. 207–213, Mar. 2005.

[153] T. Degris, M. White, and R. S. Sutton, "Off-policy actor-critic," *arXiv*, Jun. 2012.

[154] Y. Tang and S. Agrawal, "Discretizing continuous action space for on-policy optimization," *arXiv*, Jan. 2019.

[155] U. P. Networks, "Smart meter energy consumption data in London households," https://data.london.gov.uk/dataset/smartmeter-energy-use-data-in-london-households.

[156] J. E. Parsons, C. Colbert, J. Larrieu, T. Martin, and E. Mastrangelo, "Financial arbitrage and efficient dispatch in wholesale electricity markets," *MIT Center for Energy and Environmental Policy Research*, 2015.

[157] R. Li, A. J. Svoboda, and S. S. Oren, "Efficiency impact of convergence bidding in the California electricity market," *Journal of Regulatory Economics*, vol. 48, no. 3, pp. 245–284, Dec. 2015.

[158] S. Baltaoglu, L. Tong, and Q. Zhao, "Online learning of optimal bidding strategy in repeated multi-commodity auctions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 4507–4517.

[159] ——, "Algorithmic bidding for virtual trading in electricity markets," *IEEE Trans. Power Syst.*, pp. 1–1, 2018.

[160] ISO New England Inc, "Overview of new england's wholesale electricity markets and market oversight," [Online]https://www.iso-ne.com/pubs/spcl_rpts/2014/2014_market_overview_050614.pdf.

[161] ——, "ISO New England financial assurance policy," [Online]https://www.iso-ne.com/static-assets/documents/2017/09/sect_i_ex_ia.pdf.

[162] N. Yu, A. Somani, and L. Tesfatsion, "Financial risk management in restructured wholesale power markets: Concepts and tools," in *IEEE PES General Meeting*, July 2010, pp. 1–8.

[163] N. Yu, H. Sheng, and R. Johnson, "Economic valuation of wind curtailment rights," in *IEEE Power and Energy Society General Meeting (PES)*, 2013, pp. 1–5.

[164] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, "Coherent measures of risk," *Mathematical Finance*, vol. 9, no. 3, pp. 203–228, 1999.

[165] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, pp. 21–41, 2000.

[166] P. Krokhmal, J. Palmquist, and S. Uryasev, "Portfolio optimization with conditional value-at-risk objective and constraints," *Journal of Risk*, vol. 4, May 2003.

[167] EIA, "U.S. Energy Information Administration," [Online]https://www.eia.gov/.

[168] ISO New England Inc, "ISO express," [Online]https://www.iso-ne.com/markets-operations/iso-express/.

[169] PJM, "PJM electricity basics," [Online] https://learn.pjm.com/electricity-basics/market-for-electricity.aspx.

[170] S. H. Low, "Convex relaxation of optimal power flow-part I: Formulations and equivalence," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 1, pp. 15–27, Mar. 2014.

[171] M. R. Dorostkar-Ghamsari, M. Fotuhi-Firuzabad, M. Lehtonen, and A. Safdarian, "Value of distribution network reconfiguration in presence of renewable energy resources," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 1879–1888, May 2016.